



TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky
a mezioborových studií ■

Řízení asynchronního elektromotoru s polohovou zpětnou vazbou

Bakalářská práce

Studijní program: B2612 – Elektrotechnika a informatika
Studijní obor: 2612R011 – Elektronické informační a řídicí systémy
Autor práce: **Filip Satrapa**
Vedoucí práce: Ing. Martin Diblík, Ph.D.





TECHNICAL UNIVERSITY OF LIBEREC
Faculty of Mechatronics, Informatics
and Interdisciplinary Studies ■

Induction motor servodrive with position closed-loop control

Bachelor thesis

Study programme: B2612 – Electrical Engineering and Informatics
Study branch: 2612R011 – Electronic Information and Control Systems

Author: **Filip Satrapa**
Supervisor: Ing. Martin Diblík, Ph.D.



ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Filip Satrapa**
Osobní číslo: **M15000118**
Studijní program: **B2612 Elektrotechnika a informatika**
Studijní obor: **Elektronické informační a řídicí systémy**
Název tématu: **Řízení asynchronního elektromotoru s polohovou zpětnou vazbou**
Zadávající katedra: **Ústav mechatroniky a technické informatiky**

Z á s a d y p r o v y p r a c o v á n í :

1. Provedte rešerši na téma možnosti řešení polohového servopohonu s frekvenčně řízeným asynchronním elektromotorem.
2. Na základě provedené rešerše navrhnete nezbytné hw a sw řešení pro polohové řízení asynchronního elektromotoru s frekvenčním měničem pro aplikace s nízkými nároky na dynamiku polohování.
3. Zvolenou koncepci realizujte s využitím dostupného hardware.
4. Navrhnete, vytvořte a otestujte program pro PLC automat (B+R Automation) zajišťující výše uvedené řešení.
5. K vytvořenému programu zpracujte základní nápovědu.

Rozsah grafických prací: **dle potřeby dokumentace**

Rozsah pracovní zprávy: **30–40 stran**

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

- [1] **JOHN, Karl-Heinz; TIEGELKAMP, Michael. IEC 61131-3: programming industrial automation systems : Concepts and Programming Languages, Requirements for Programming Systems, Decision-Making Aids . 2nd ed. Springer, 2010. ISBN 9783642120145.**
- [2] **B+R Automatizace. Automation Training Materials, 2009.**
- [3] **SIEMENS A.G. Siemens Micromaster 440: Operating Instructions, Reference Manual. 2001.**
- [4] **Siemens A.G. Sinamics G110 120W - 3kW. Návod k obsluze, příručka uživatele. Vydání 04/2003.**

Vedoucí bakalářské práce:

Ing. Martin Diblík, Ph.D.

Ústav mechatroniky a technické informatiky

Datum zadání bakalářské práce: **10. října 2018**

Termín odevzdání bakalářské práce: **30. dubna 2019**

L.S.

prof. Ing. Zdeněk Plíva, Ph.D.
děkan

doc. Ing. Milan Kolář, CSc.
vedoucí ústavu

V Liberci dne 10. října 2018

Prohlášení

Byl jsem seznámen s tím, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé bakalářské práce pro vnitřní potřebu TUL.

Užiji-li bakalářskou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Bakalářskou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím mé bakalářské práce a konzultantem.

Současně čestně prohlašuji, že tištěná verze práce se shoduje s elektronickou verzí, vloženou do IS STAG.

Datum:

Podpis:

Abstrakt

Práce se zabývá řešením problému polohování s asynchronním elektromotorem. Byla provedena rešerše praktických řešení tohoto problému. Následně byl definován konkrétní způsob realizace, využívající řízení asynchronního pohonu na konstantní rychlost se zpětnou polohovou vazbou. Byl vytvořen programový blok pro ovládání takového typu pohonu, přičemž blok je určen k integraci do komplexnějších řídicích programů. Kód ovládacího programu je napsaný ve strukturovaném textu.

Klíčová slova: PLC, M440, B+R, Siemens, elektromotor, polohování s kladnou zpětnou vazbou

Abstract

The thesis is about to create a solution to the problem of positioning with an asynchronous electric motor. A search has been made to find out about already existing solutions. Consequently, we chose a specific method for realizing the issue. In our case, we used asynchronous drive control at a constant speed with position feedback. Then we have created a program block for controlling our application. The block is designed to be integrated into a more complex solution. A language used to program the program book was structured text.

Keywords: PLC, M440, B+R, Siemens, electromotor, positioning with positive feedback

Obsah

Seznam symbolů, zkratk a termínů	10
1 Úvod	11
2 Rešerše	11
3 Frekvenční řízení asynchronního motoru	14
3.1 Asynchronní motor	14
3.2 Způsoby řízení motoru	14
3.3 Frekvenční měnič	15
4 Základní HW požadavky	15
4.1 Výchozí předpoklady	15
4.2 Požadavky na dráhu	16
4.3 Blokové schéma zapojení úlohy	16
5 Použitý hardware pro realizaci úlohy	17
5.1 Frekvenční měnič	18
5.1.1 Blokové schéma měniče	18
5.2 Inkrementální snímač LARM IRC305	20
5.3 Multifunkční signálová digitální karta X20DS1119	20
6 Program pro řízení pohonu	21
6.1 Základní koncepce a struktura	21
6.2 Blok „Autocontrol“	23
6.3 Blok „Výpočet difference“	25
6.4 Blok „Home“	26
6.5 Blok „RMP Scan“	27
6.6 Blok „Scan“	30
7 Návod	31
7.1 Seznam důležitých proměnných pro uživatele:	32
7.2 Nastavení měniče	33
7.2.1 Obecné nastavení měniče	33
7.2.2 Nastavení měniče M440	34
7.3 Zapojení úlohy	34
7.4 Návod ke spuštění jednotlivých bloků	37

7.4.1	Prvotní spuštění - inicializace.....	38
7.4.2	Blok pro automatické nájezdy na bod	38
7.4.3	Blok pro sken dráhy	38
7.4.4	Blok pro nájezd na referenční čidlo.....	38
7.4.5	Blok pro sken délky ramp	38
8	Ověření funkčnosti	39
8.1	Postup a způsob měření:	39
8.2	Tabulka naměřených hodnot ramp	39
8.3	Grafy naměřených hodnot	40
8.4	Měření ramp vysoké rychlosti	41
8.5	Polohování s nastavenými rampami	42
8.6	Graf nájezdu na žádanou pozici.....	44
8.7	Výsledky měření.....	44
9	Možnosti budoucí modifikace programu.....	44
9.1	Modifikace pro rotační osu.....	44
9.2	Modifikace pro zápornou polohu čidla	45
9.3	Zakázání polohování pro malé změny polohy	45
10	Závěr:	45
11	Seznam použité literatury	46
12	Přílohy	

Seznam obrázků

OBRÁZEK 1 PŘÍKLAD ŘÍZENÉHO SERVOMOTORU OD FIRMY SIEMENS [8].....	12
OBRÁZEK 2 STRUKTURA SYSTÉMU OD ALLEN BRADLEY PŘEVZATO Z [7]	12
OBRÁZEK 3 SCHÉMA INTREGROVANÉ FUNKCE PID REGULÁTORU OD FIRMY ALLEN BRADLEY PŘEVZATO Z [7]	12
OBRÁZEK 4 ŘEŠENÍ NÁJEZDU NA REFERENČNÍ ČIDLO AS [6].....	13
OBRÁZEK 5 NAJÍŽDĚNÍ NA REFERENČNÍ BOD OD FIRMY SIEMENS PŘEVZATO Z [8].....	13
OBRÁZEK 6 POŽADAVKY NA DRÁHU.	16
OBRÁZEK 7 BLOKOVÉ SCHÉMA ÚLOHY	16
OBRÁZEK 8 POUŽITÉ PLC TYP X20CP1585 PŘEVZATÉ Z EXISTUJÍCÍ ÚLOHY, KDE PŮVODNÍ MODULY BYLY ODPOJENY A NAHRAZENY NAŠIMI	17
OBRÁZEK 9 POUŽITÝ FREKVENČNÍ MĚNIČ M440 ZAPOJENÝ V LABORATORNÍ ÚLOZE A DETAIL NA SVORKOVNICI.....	18
OBRÁZEK 10 BLOKOVÉ SCHÉMA MĚNIČE M440 PŘEVZATO Z [1].....	19
OBRÁZEK 11 POUŽITÝ SENZOR LAMR IRC305 NAMONTOVANÝ NA HŘÍDELI MOTORU.....	20
OBRÁZEK 12 PINOUT MODULU DS1119 PŘEVZATO Z [4]	21
OBRÁZEK 13 CELKOVÝ NÁHLED PROGRAMU	22
OBRÁZEK 14 VÝVOJOVÝ DIAGRAM BLOKU AUTOCONTROL.....	23
OBRÁZEK 15 RYCHLOSTNÍ PROFIL BLOKU AUTOCONTROL PŘI VYSOKÉ RYCHLOSTI.....	24
OBRÁZEK 16 RYCHLOSTNÍ PROFIL BLOKU AUTOCONTROL PŘI NÍZKÉ RYCHLOSTI	24
OBRÁZEK 17 VÝVOJOVÝ DIAGRAM BLOKU PRO VÝPOČET DIFERENCE.....	25
OBRÁZEK 18 VÝVOJOVÝ DIAGRAM BLOKU PRO NÁJEZD NA HOME POZICI.....	26
OBRÁZEK 19 RYCHLOSTNÍ PROFIL BLOKU PRO NÁJEZD NA HOME POZICI	27
OBRÁZEK 20 VÝVOJOVÝ DIAGRAM BLOKU PRO MĚŘENÍ RAMP	28
OBRÁZEK 21 RYCHLOSTNÍ PROFIL BLOKU PRO MĚŘENÍ RAMP PŘI VYSOKÉ RYCHLOSTI.....	29
OBRÁZEK 22 RYCHLOSTNÍ PROFIL BLOKU MĚŘENÍ RAMP PŘI NÍZKÉ RYCHLOSTI	29
OBRÁZEK 23 VÝVOJOVÝ DIAGRAM BLOKU SKEN	30
OBRÁZEK 24 RYCHLOSTNÍ PROFIL BLOKU PRO SKEN DRÁHY	31
OBRÁZEK 25 ZOBRAZENÍ PROMĚNNÉ PÁSMO NÍZKÉ RYCHLOSTI.....	33
OBRÁZEK 26 SIGNÁLOVÉ ZAPOJENÍ ÚLOHY	35

Seznam grafů

GRAF 1 MĚŘENÍ RAMP PŘI NÍZKÉ RYCHLOSTI A NEGATIVNÍM SMĚRU	40
GRAF 2 MĚŘENÍ RAMP PŘI NÍZKÉ RYCHLOSTI A POZITIVNÍM SMĚRU	40
GRAF 3 MĚŘENÍ RAMP PŘI VYSOKÉ RYCHLOSTI A POZITIVNÍM SMĚRU	41
GRAF 4 MĚŘENÍ RAMP PŘI VYSOKÉ RYCHLOSTI A NEGATIVNÍM SMĚRU	41
GRAF 5 MĚŘENÍ PŘESNOSTI POLOHOVÁNÍ PŘI NEGATIVNÍM SMĚRU	42
GRAF 6 MĚŘENÍ PŘESNOSTI POLOHOVÁNÍ PŘI POZITIVNÍM SMĚRU.....	43
GRAF 7 VÝVOJ POZICE PŘI NÁJEZDU NA ŽÁDANÝ BOD	44

Seznam tabulek

TABULKA 1 ZÁKLADNÍ DATA SENZORU LAMR IRC 305 PŘEVZATO Z [5]	20
TABULKA 2 DŮLEŽITÉ PROMĚNNÉ.....	32
TABULKA 3 NASTAVENÍ HODNOT FIXNÍCH FREKVENCÍ.....	34
TABULKA 4 NASTAVENÍ DIGITÁLNÍCH VÝSTUPŮ.....	34
TABULKA 5 NASTAVENÍ PARAMETRŮ MĚNIČE M440.....	34
TABULKA 6 KOMBINACE VÝSTUPŮ PRO ZADÁNÍ FREKVENCÍ.....	34
TABULKA 7 SEZNAM PROPOJENÍ SVORKOVNIC	36
TABULKA 8 NAMĚŘENÉ HODNOTY RAMP.....	39
TABULKA 9 POLOHOVÁNÍ V NEGATIVNÍM SMĚRU.....	42
TABULKA 10 POLOHOVÁNÍ V POZITIVNÍM SMĚRU.....	43

Seznam symbolů, zkratk a termínů

DIN	–	digitální vstup
DOU	–	digitální výstup
f_{\max}	–	maximální frekvence
ot	–	otáčka
AS	–	automation studio
Pos	–	pozitivní
Neg	–	negativní
IRC	–	inkrementální rotační snímač
inc	–	inkrement
NO	–	normally open
NC	–	normally close
s [inc]	–	poloha pohonu
v [inc/s]	–	rychlost
v_f [inc/s]	–	vysoká rychlost
v_c [inc/s]	–	nízká rychlost
n_s [ot/s]	–	synchronní otáčky
p [-]	–	počet pólových dvojic
f [Hz]	–	frekvence

1 Úvod

Cílem této práce bylo navrhnout program pro polohování pomocí frekvenčně řízeného maximálně dvourychlostního asynchronního elektromotoru s nízkou dynamikou. Jedná se o relativně obecný problém, jehož řešení zpracovává mnoho výrobců. Tato řešení mohou být relativně nákladná. Navíc pokud se zvolí, zpravidla se předpokládá použití hardwarových komponent přímo od daného výrobce. Opět to nemusí být zcela ekonomicky výhodné, na druhou stranu výrobce zaručuje jejich plnou kompatibilitu.

Zde se zabýváme tvorbou programu, který je otestován na hardwaru složeném od několika výrobců – PLC od firmy B&R zde řídí měnič od firmy Siemens. Vytvořený program se skládá ze dvou podprogramů, kde každý z nich obsahuje několik základních bloků a podpůrné funkce. Rozdělení do dvou podprogramů je hlavně z toho důvodu, že u některých bloků je důležité, aby operovali v rychlejším cyklu.

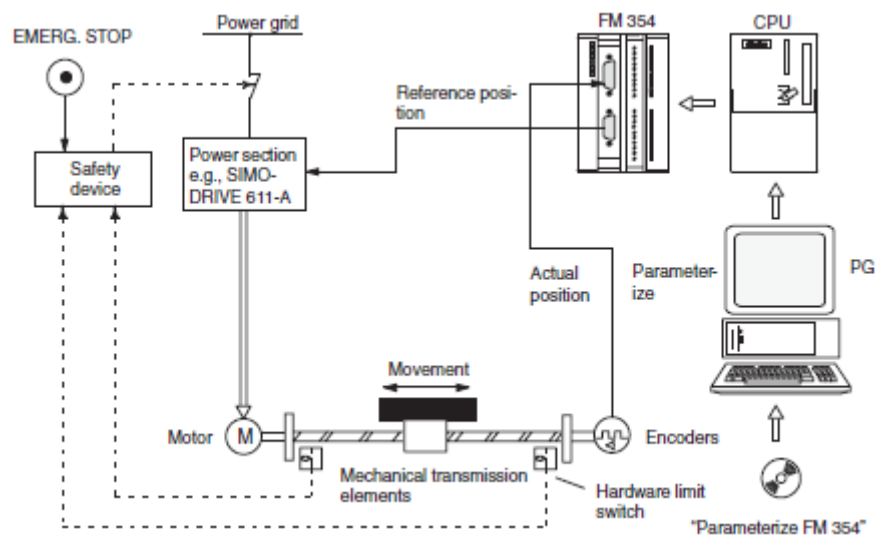
2 Rešerše

Pro rozbor řešení tohoto problému, jsme využili již hotových řešení ostatních firem. Zvolili jsme firmy Omron [9], Siemens [8], Mitsubishi [10], Allen Bradley [7] a TECO. Všechny tyto firmy nabízejí hardware, který má integrované funkce pro řešení našeho problému. Firmy nabízejí toto řešení v několika HW variantách, a to buď jako přídatnou měřicí kartu, nebo jako nanoPLC například MAC00-R1.

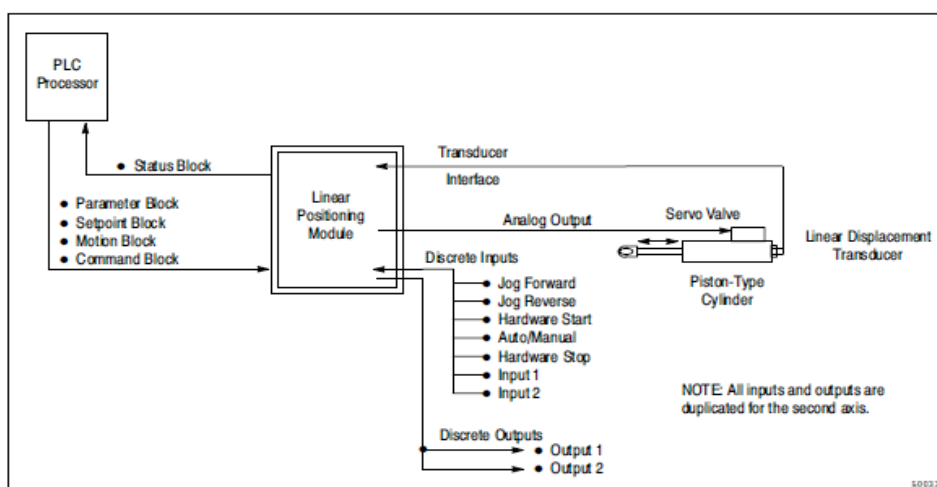
Řešení od jednotlivých firem se příliš neliší, zpravidla se jedná o drobné úpravy a jisté přídatné funkce. Například firma Allen Bradley nabízí přednastavenou možnost pro imperiální systém jednotek a integrovaný softwarový PID regulátor (Obrázek 3). Struktura systému od firmy Allen Bradley viz Obrázek 2. Příklad struktury pohonu od firmy Siemens viz Obrázek 1.

I přesto že každý výrobce má lehce odlišnou strukturu, základní funkce jsou principiálně stejné. Jedná se například o diskrétní řízení, pohybování mezi nastavenými setpointy, pásma necitlivosti a podobně.

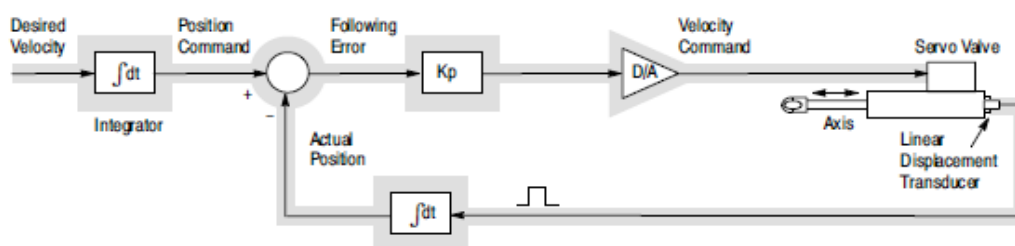
Mnoho důležitých informací o provedení a funkčnosti různých částí programu (jako nájezd na home pozici, viz Obrázek 5) lze najít i v nápovědě AS [6], kde jsou funkce, jak popsány, tak doplněny grafy průběhů.



Obrázek 1 Příklad řízeného servomotoru od firmy Siemens [8] str 24



Obrázek 2 Struktura systému od Allen Bradley převzato z [7] str 15



Obrázek 3 Schéma integrované funkce PID regulátoru v pozicovací smyčce od firmy Alen Bradley převzato z [7] str 19

```

edge_sw = ncNEGATIVE
trigg_dir = ncNEGATIVE
fix_dir = ncOFF

```

```

edge_sw = ncNEGATIVE
trigg_dir = ncPOSITIVE
fix_dir = ncOFF

```

```

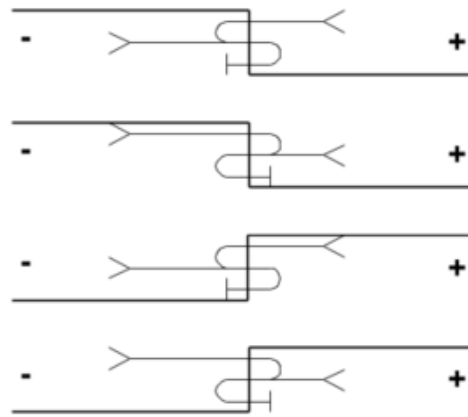
edge_sw = ncPOSITIVE
trigg_dir = ncNEGATIVE
fix_dir = ncOFF

```

```

edge_sw = ncPOSITIVE
trigg_dir = ncPOSITIVE
fix_dir = ncOFF

```



Obrázek 4 Řešení nájezdu na referenční čidlo AS [6]

Jeden z příkladů řešení najetí na referenční bod z automation studio. Jak je vidět tento způsob zohledňuje jakou hranu referenčního musíme dostat a z jakého směru se máme přiblížit. Další parametry jako je fixní směr pohybu lze najít v nápovědě AS [6].

Triggering of movement, direction for synchronization (R)	Type of reference-point approach	Sequence of motions (reference point offset = 0) V_A – referencing velocity V_R – reducing velocity
R+ ("edge-controlled") or Start	1st situation Current-sourcing pattern zero or zero pulse external, to right of RPS	
	2nd situation Current-sourcing pattern zero or zero pulse external, to left of RPS	
	3rd situation RPS centered (no zero pulse necessary)	
	4th situation RPS edge (no zero pulse necessary)	
R- ("edge-controlled") or Start	1st situation see above	equals R+ 2nd situation mirrored
	2nd situation see above	equals R+ 1st situation mirrored
	3rd situation RPS centered (no zero pulse necessary)	equals R+ 3rd situation mirrored
	4th situation RPS centered (no zero pulse necessary)	equals R+ 4th situation mirrored

Obrázek 5 Najíždění na referenční bod od firmy Siemens převzato z [8] str 226.

3 Frekvenční řízení asynchronního motoru

3.1 Asynchronní motor

Jedná se o elektrický stroj složený ze statoru a rotoru. Kde stator je napájen střídavým proudem, to vytváří střídavé magnetické pole, které indukuje napětí do rotoru. Tím se vytváří síla, která v kombinaci se statorovým polem roztáčí rotor. To že je motor asynchronní, znamená, že jeho otáčky jsou nižší, než otáčky synchronní, ty jsou dané následujícím vztahem.

$$n_s = \frac{60 \cdot f}{p}. \quad (1)$$

Kde f je napájecí frekvence a p počet pólových dvojic. Vzhledem k principu asynchronního motoru by při dosažení synchronních otáček došlo k tomu, že se do rotoru přestane indukovat napětí. Vlivem skluzu který definujeme jako zpoždění rotorového pole za statorovým jsou však otáčky nižší. Jejich výpočet je dán následujícím vztahem.

$$n = n_s - \frac{\text{skluz} \cdot n_s}{100}. \quad (2)$$

Kde *skluz* je udán v procentech. Ze vztahu pro výpočet otáček vyplývají i způsoby řízení asynchronního motoru.

3.2 Způsoby řízení motoru

Pro řízení asynchronního motoru je k dispozici několik možností. V naší aplikaci využíváme řízení pomocí změny frekvence napájení. Jeho výhodou je možnost regulace otáček ve velkém rozsahu a také to, že nevyžaduje specifikou konstrukci motoru. Nevýhodou je že pro jeho realizaci je třeba zapojit externí frekvenční měnič. Tímto způsobem lze motor i brzdít, a to postupným snižováním frekvence. Tento způsob je využíván i v našem případě.

Další možností řízení otáček je přepínatelným počtem pólových dvojic. Největší nevýhodou tohoto způsobu je, že motor musí být specificky konstruován. Navíc nelze měnit frekvenci libovolně ale pouze skokově.

Poslední možností je řízení skluzem. Toto řízení je však možné pouze na velmi malém rozsahu otáček a jedná se o ztrátové řízení, v praxi téměř nepoužívané.

Vzhledem k tomu. Že naše aplikace počítá s maximálně dvou rychlostním pohonem, tak se nabízí i možnost využití motoru s přepínatelným počtem pólových dvojic. Námi zvolený postup je však využití standardního asynchronního motoru řízeného frekvenčním měničem, hlavně proto že se jedná o nejtypičtější a nejuniverzálnější způsob řízení.

Pro zvolený motor je nutno brát ohled na jeho chlazení. Námi použitý motor je motor standardní, u kterého využíváme nižší otáčky, než na jaké je stavěn. V našem případě se nejedná o problém, protože motor neběží dlouhodobě. Proto jsme mohli tuto problematiku zanedbat. Pokud by však byl tento motor použit v dlouhodobém chodu, při těchto nízkých otáčkách, hrozilo by jeho přehřátí. Pokud bychom tento problém museli brát v potaz, je nutno vybrat motor který se na těchto nízkých otáčkách uchladí vlastním chlazením, nebo využít motor s chlazením externím.

3.3 Frekvenční měnič

Jedná se o zařízení sloužící k vytvoření napájení o alternativní frekvenci, než je frekvence sítě. Velmi zjednodušeně můžeme říct, že se skládá ze tří základních bloků. První je usměrňovač vstupního napětí. Druhý je stejnosměrný meziobvod a poslední částí je střídač. Měniče lze rozdělit do dvou základních kategorií, a to je vektorové řízení a skalární řízení. Vektorové se vyznačují komplikovanější konstrukcí, která umožňuje provádět i náročnější funkce, zároveň jsou tím také dražší. Pro potřeby naší aplikace je skalární měnič zcela dostačující.

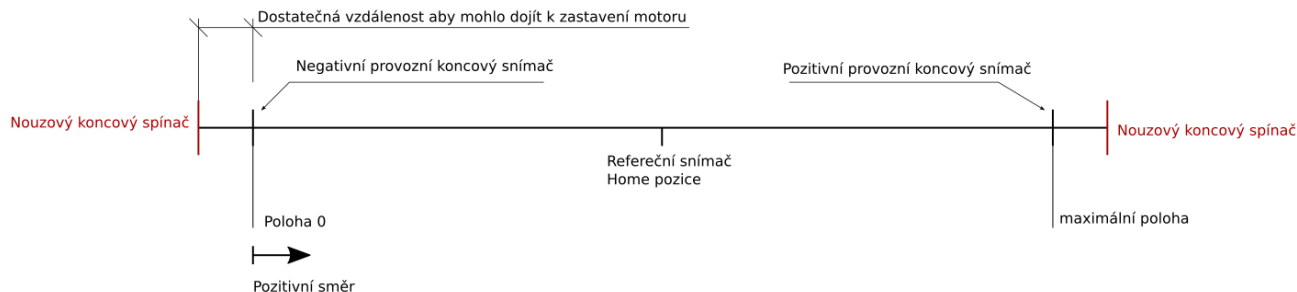
4 Základní HW požadavky

4.1 Výchozí předpoklady

- 1.1 Základní předpoklad aplikace je, že počítáme s pohonem s jednou nebo max. dvěma konstantními rychlostmi. V praxi toho lze dosáhnout třemi základními způsoby:
 - a. Asynchronní motor s přepínatelným počtem pólových dvojic.
 - b. Asynchronní motor napájen přes frekvenční měnič. U něj je pak možné řešit přepínání rychlostí dvěma základními způsoby:
 - i. Přednastavené frekvence, mezi nimiž přepínáme kombinací digitálních vstup. Tento způsob je využit v naší aplikaci.
 - ii. Analogový vstup, u kterého se využijí pouze určité hodnoty.
 - c. Motor napájený přímo ze sítě ten je samozřejmě jedno rychlostní.
- 1.2 Informace o aktuální pozici je získána pomocí snímače natočení, který je nainstalovaný na hřídel motoru nebo na koncový mechanismus. Zpravidla je využit optický inkrementální snímač natočení, ten je využit i naší úlohou.
- 1.3 Počítáme se čtyřmi koncovými snímači, kde dva jsou provozní. S nimi operuje program, jeden v pozitivním směru, druhý v negativním. Jejich funkce je zamezení dalšího pohybu daným směrem. Další dva koncové snímače jsou nouzové. Referenční čidlo být může a nemusí, pokud není, lze zadat pozici softwarově a stále tak lze využít funkci home. Obrázek 6 znázorňuje jejich předpokládané rozložení. Na detekčních hranách čidel nezáleží, program si je následně upraví, aby se vždy chovaly jako rozpínací.
- 1.4 Osa pohybu musí být chápána jako lineární, na rotační osu program není adaptován.

4.2 Požadavky na dráhu

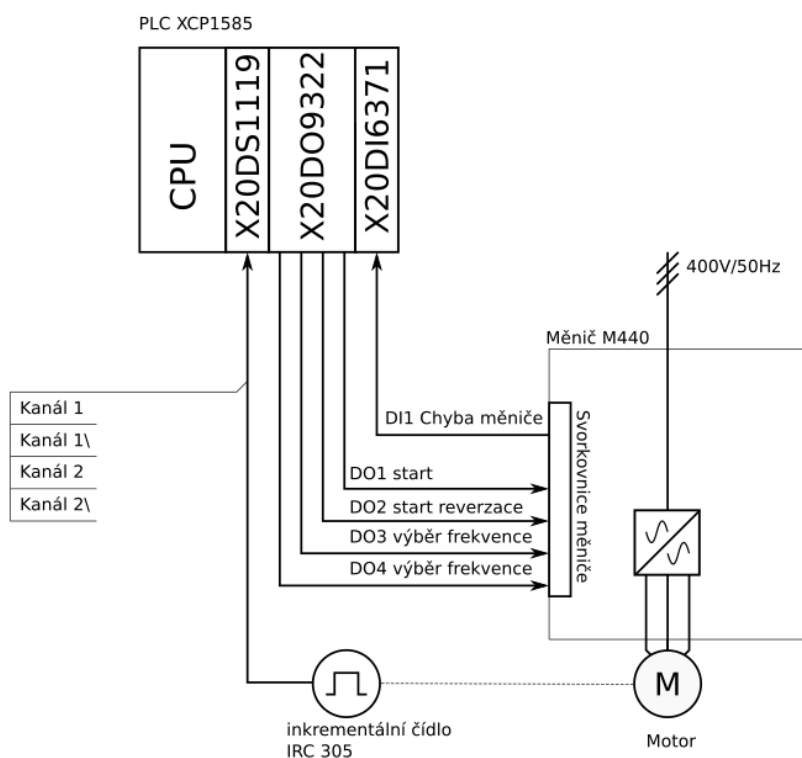
Obrázek 6 zobrazuje umístění snímačů na ose, a základní informace jako pozitivní směr pohybu a umístění nuly.



Obrázek 6 Požadavky na dráhu.

4.3 Blokové schéma zapojení úlohy

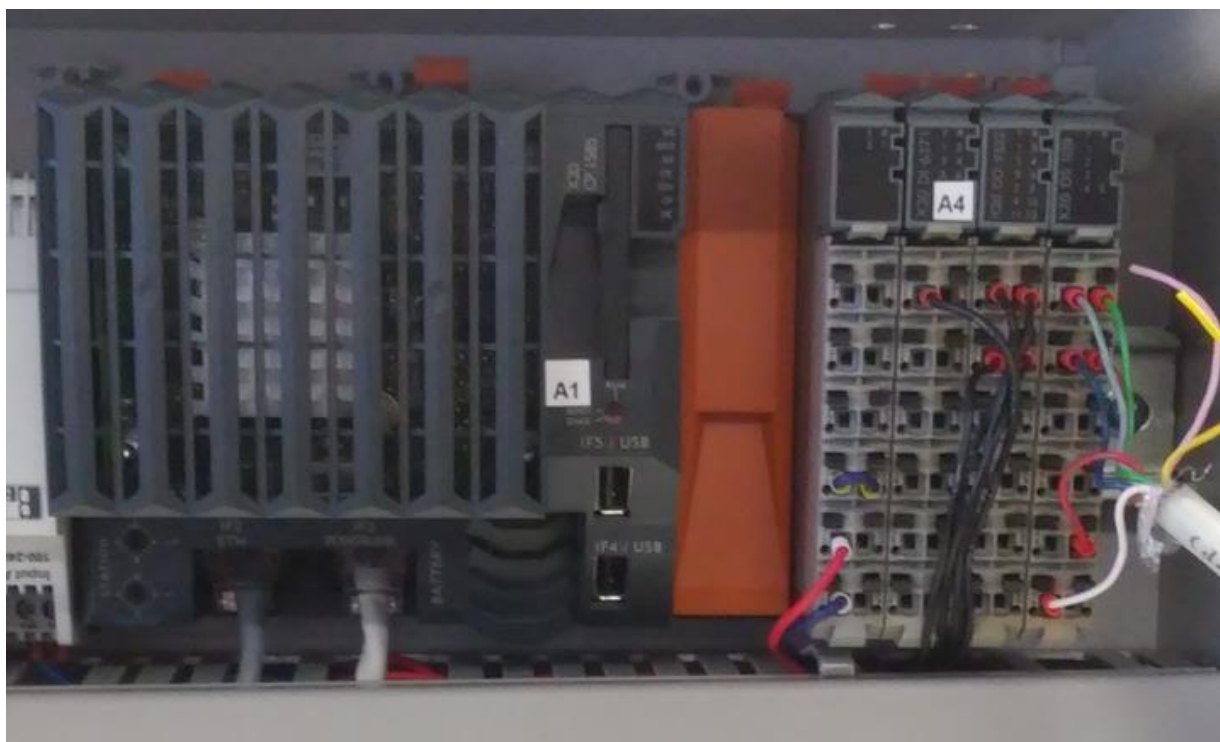
Obrázek 7 na kterém je vidět schéma, které znázorňuje principiální zapojení naší úlohy. Konkrétní realizaci pak znázorňuje Obrázek 26 a přesné zapojení svorkovnice Tabulka 7. Koncové snímače v naší realizované aplikaci byly pouze virtuální. Proto nejsou uvedeny v blokovém schématu, jinak by ale byly napojeny na modul X20DI6371.



Obrázek 7 Blokové schéma úlohy

5 Použitý hardware pro realizaci úlohy

Pro účely ovládání pohonu bylo nutné použít PLC, které disponuje alespoň čtyřmi digitálními výstupy a jedním digitálním vstupem. Řízení rychlosti pohonu je řešeno pomocí fixních frekvencí, proto není potřeba analogový signál. Dva digitální výstupy pokryjí ovládání (vpřed, vzad), zbylé dva jsou použity pro nastavení frekvencí. Digitální vstup je použit pro signalizaci chyby frekvenčního měniče.



Obrázek 8 Použité PLC typ X20CP1585 převzaté z existující úlohy, kde původní moduly byly odpojeny a nahrazeny našimi

Pro naši úlohu byl zvolen dostupný hardware z vybavení ústavu. PLC bylo převzato z jiné úlohy, kde jsme pouze přepojili moduly na námi potřebné a zbytek úlohy odpojili. Konkrétně se jednalo o PLC X20CP1585 se třemi rozšiřujícími moduly X20DS1119 (pro vyhodnocení inkrementálního čidla), X20DO9233 (modul digitálních výstupů) a X20DI6371 (modul digitálních vstupů).

Frekvenční měnič ve spojení s motorem byl rovněž realizován jako učební pomůcka s ručním řízením. To jsme přepojili na ovládání z PLC modulů a na hřídel motoru jsme připojili IRC čidlo.

5.1 Frekvenční měnič



Obrázek 9 Použitý frekvenční měnič M440 zapojený v laboratorní úloze a detail na svorkovnici

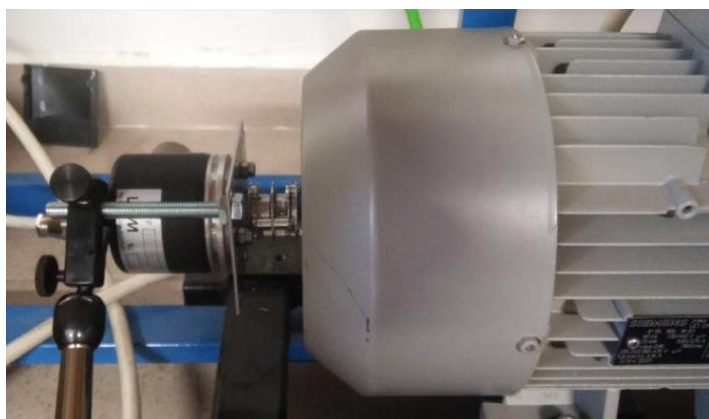
Pro řízení asynchronního elektromotoru byl použit frekvenční měnič Siemens Micromaster M440 s ovládacím BOP panelem. Jedná se o model frekvenčního měniče s vektorovým řízením. Měnič není uzpůsoben na využití rekuperačního brzdění.

5.1.1 Blokové schéma měniče

Shrnutí důležitých parametrů:

- 6 digitálních vstupů
- 6 digitálních výstupů
- 2 analogové vstupy
- 2 analogové výstupy

5.2 Inkrementální snímač LARM IRC305



Obrázek 11 Použitý senzor LARM IRC305 namontovaný na hřídeli motoru

Pro snímání polohy servopohonu resp. natočení hřídele motoru byl použit inkrementální optický snímač LARM IRC 305.

„Inkrementální rotační snímače typu IRC s LED-diodou v osvětlovači a ve standardním průmyslovém provedení převádí rotační pohyb na elektrické signály pomocí fotoelektrického snímání rastrů dvou skleněných prvků (statoru a rotoru). Jsou určeny pro zprostředkování elektrické informace o vzájemné poloze dvou mechanických celků, úhlovém natočení nebo rotačních pohybech. Typické použití snímačů typu IRC je ve spojení s číslicovými indikacemi nebo řídicími systémy. S výhodou je lze použít i v jiných zařízeních kde je potřebná vysoká přesnost a spolehlivost odměřování.“

Překlad z manuálu čidla [5] str 1

Tabulka 1 Základní data senzoru LARM IRC 305 převzato z [5] str 1

Elektrical data	IRC 3×5
Supply voltage U_n [V]	$5 \pm 5\%$
Supply voltage OC U_o [V]	-
Supply current max. I_n [mA]	100
Output frequency max. F_o [kHz]	200
Output max. I_o [mA]	± 20

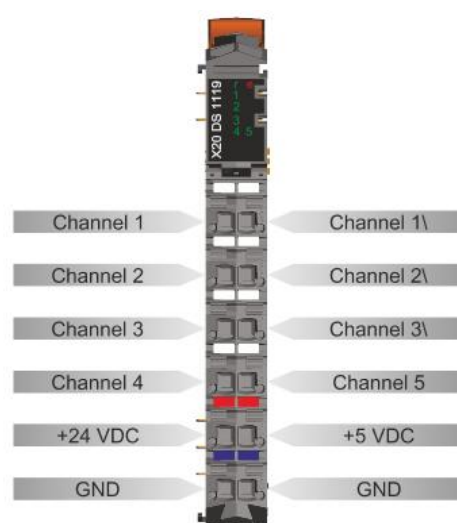
Výstup čidla je troj-kanálový, pro námi použitý vyhodnocovací modul je využit pouze první a druhý kanál, a to jak regulární, tak negovaný. Pro vyhodnocení signálu byl využit nastavný modul X20DS1119, jedná se o multifunkční modul pro vyhodnocení digitálních signálů.

5.3 Multifunkční signálová digitální karta X20DS1119

Jedná se o multifunkční digitální signálový procesor, jeho použití lze softwarově definovat. Kartu lze použít pro vyhodnocení rychlých digitálních vstupů, jako univerzální čítač, jako generátor lineárních pohybových pulsů atd. (viz.[4]). Pinout modulu viz Obrázek 12

- 3× 5V digitální kanál konfigurovatelný na vstup nebo výstup jenž jsou schopny zpracovat signál o frekvenci až 600kHz
- 2× vstupní digitální 24V kanály

Schopnost vyhodnotit signál s frekvencí do 600kHz je pro naši úlohu, kde se počítá s nízkou dynamikou více než dostačující. Díky možnosti snímání signálu s takto vysokou frekvencí je možné čidlo umístit přímo na hřídel motoru. To nám poskytuje alternativu k možnosti montáže čidla přímo na koncový mechanismus. Čidlo má 5000 impulzů na otáčku, karta samotná potom počet pulzů násobí čtyřmi. Na jednu otáčku je tedy počítáno s 20 000 pulzy. Modul pracuje v režimu čítače.



Obrázek 12 Pinout modulu DS1119 převzato z [4] str 4

6 Program pro řízení pohonu

6.1 Základní koncepce a struktura

Program je vytvořen tak, aby byla možná jeho mírná parametrizace rukou uživatele na danou úlohu, kde má být použit. Například lze zvolit patřičný typ koncového snímače nebo typ nájezdu na referenční čidlo.

Co se jednotek programu týče program je navržen, aby pracoval v pulzech inkrementálního čidla. Přepočty na fyzikální jednotky nejsou řešeny. **Velmi důležité je také že polohování je v absolutních souřadnicích a na lineární ose.** I přes to, že program je schopen provádět některá měření sám, pokud je uživatel chce nadefinovat sám, tak to program umožňuje.

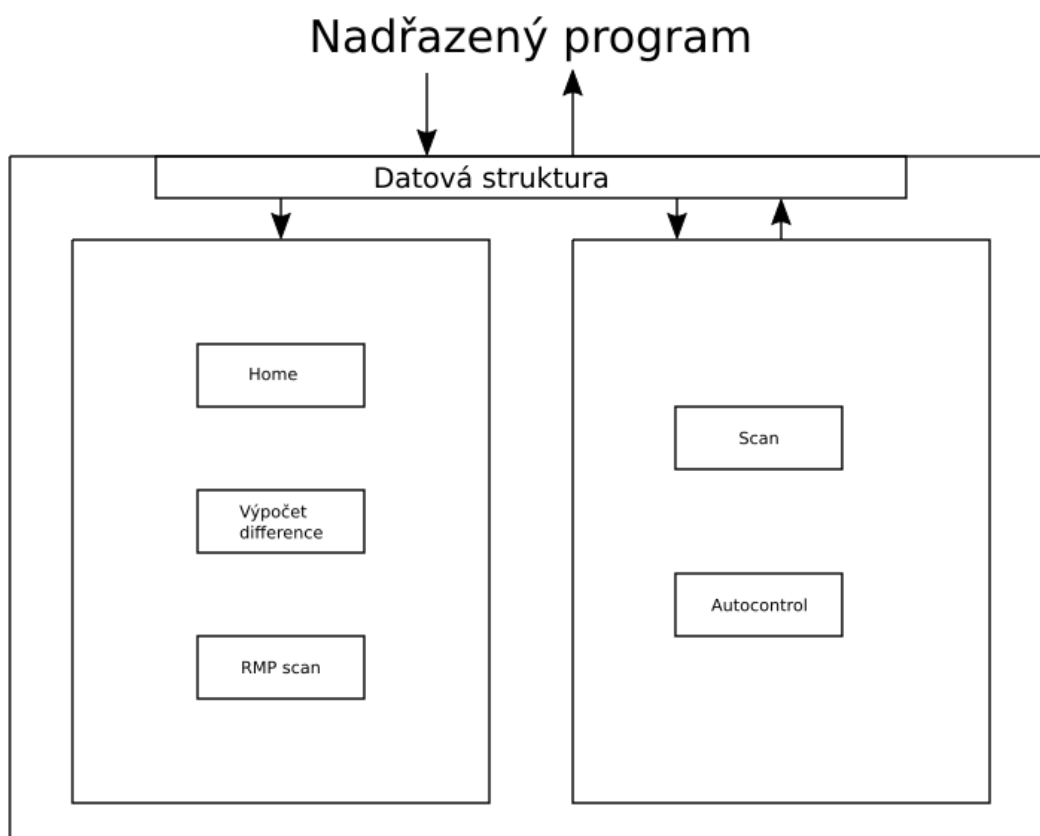
Obrázek 13 popisuje základní struktura programu. Hlavním blokem programu je blok pro automatické řízení (*Autocontrol*), na ten jsou navázány další bloky programu a rozšíření o další funkce. Pro hlavní bloky programu jsou vytvořeny vývojové diagramy. U bloků, které to umožňují je vytvořen rychlostní profil.

Pro prvotní spuštění programu je zde naprogramováno několik funkcí, které napomohou hladkému chodu programu bez většího vnějšího zásahu uživatele.

Jedná se o tyto funkce (bloky):

1. První funkce je *Scan* (Obrázek 23). Tato funkce má za úkol zmapování dráhy úlohy – provádí měření délky dráhy mezi záporným a kladným provozním koncovým snímačem. Současně během pohybu program sleduje, kdy dostane pulz referenčního čidla. Tuto hodnotu program ukládá.
2. Další pomocný blok je použitý k určení délek ramp motoru (*RMP Scan* Obrázek 20). Program provede měření ramp motoru podle parametrů zvolených uživatelem. Vývojový diagram této funkce je lehce zjednodušený, je v něm vysvětlen princip. Program rozlišuje a ukládá hodnoty do různých polí podle směru, kterým se motor otáčí. Princip je všude stejný a zařazením těchto variací by došlo jenom k rozvětvení diagramu a zhoršení přehlednosti.

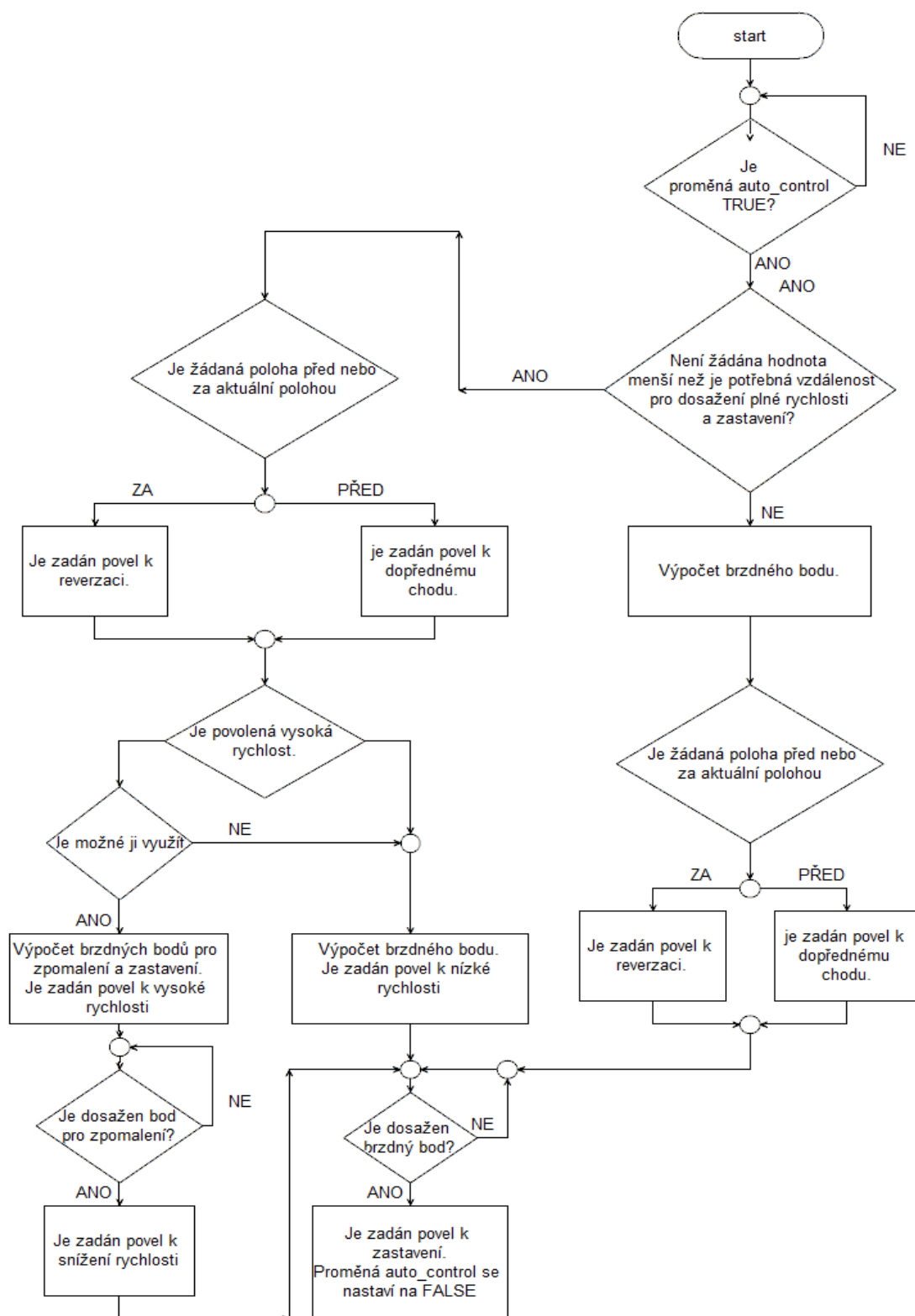
Obrázek 13 je celkový náhled programu, můžeme tam vidět základní strukturu. Program je dělen do dvou podprogramů, kde každý z nich potom obsahuje několik dalších větších bloků plus další funkce které zajišťují plynulý chod. Komunikace s nadřazeným programem či uživatelem je potom realizována přes hlavní datovou strukturu.



Obrázek 13 Celkový náhled programu

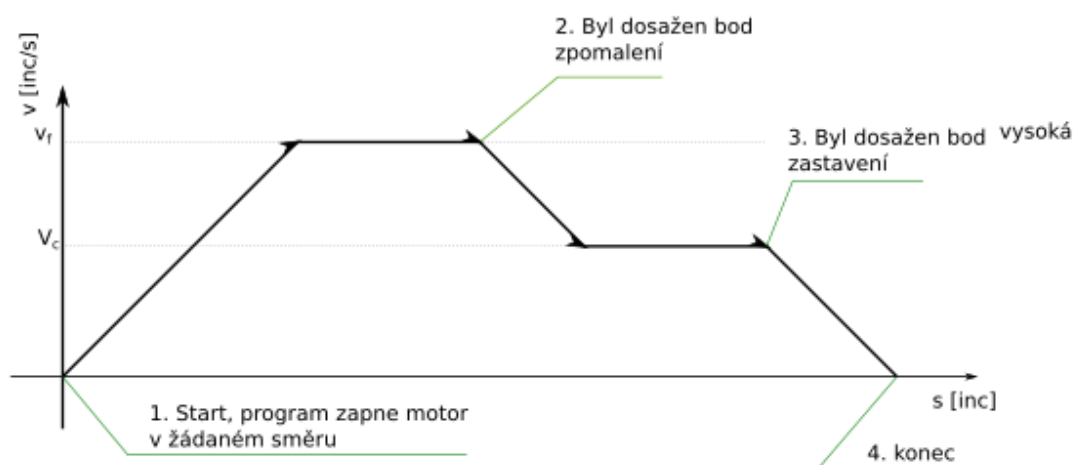
6.2 Blok „Autocontrol“

Tento blok realizuje vlastní automatické polohování pohonu na žádanou pozici. Polohování může probíhat ve dvou „režimech“, které jsou znázorněny na obrázku 15 a 16. Pokud je proměnná

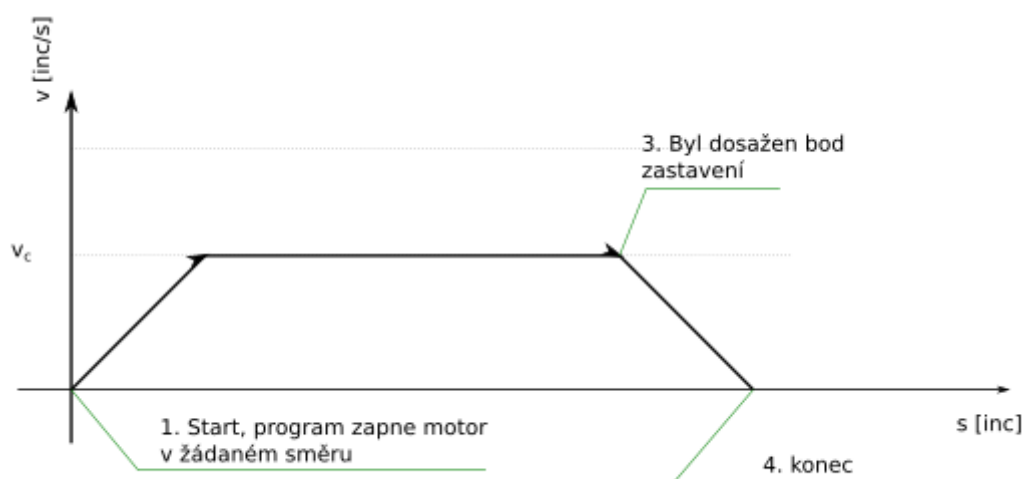


Obrázek 14 Vývojový diagram bloku autocontrol

M440.CMD.fast_speed_enb = TRUE, potom platí Obrázek 15, pokud je M440.CMD.fast_speed_enb = FALSE potom platí Obrázek 16.



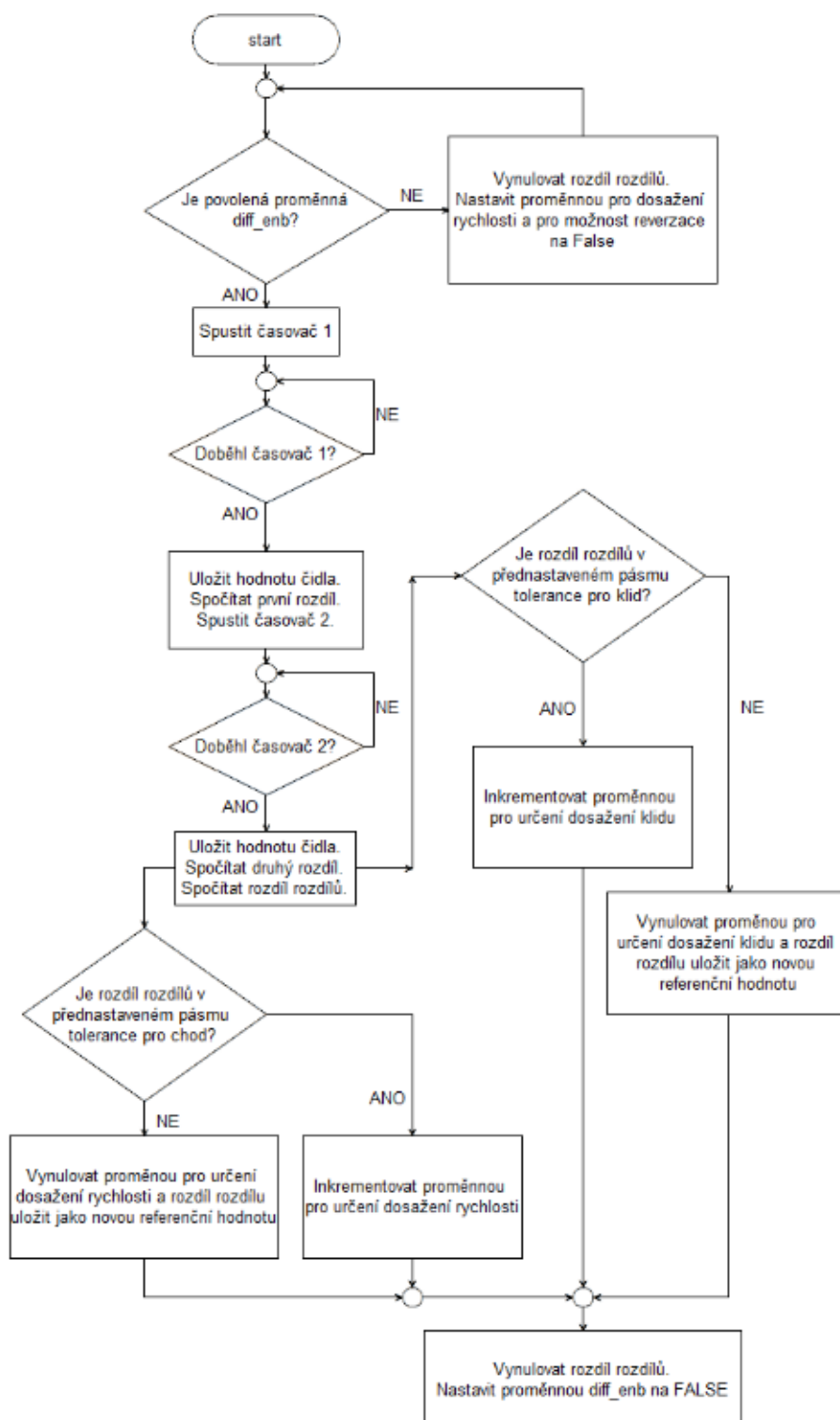
Obrázek 15 Rychlostní profil bloku autocontrol při vysoké rychlosti



Obrázek 16 Rychlostní profil bloku autocontrol při nízké rychlosti

6.3 Blok „Výpočet difference“

Tento blok pro uživatele není příliš důležitý, sám uživatel ho ani nespouští, jedná se o interní

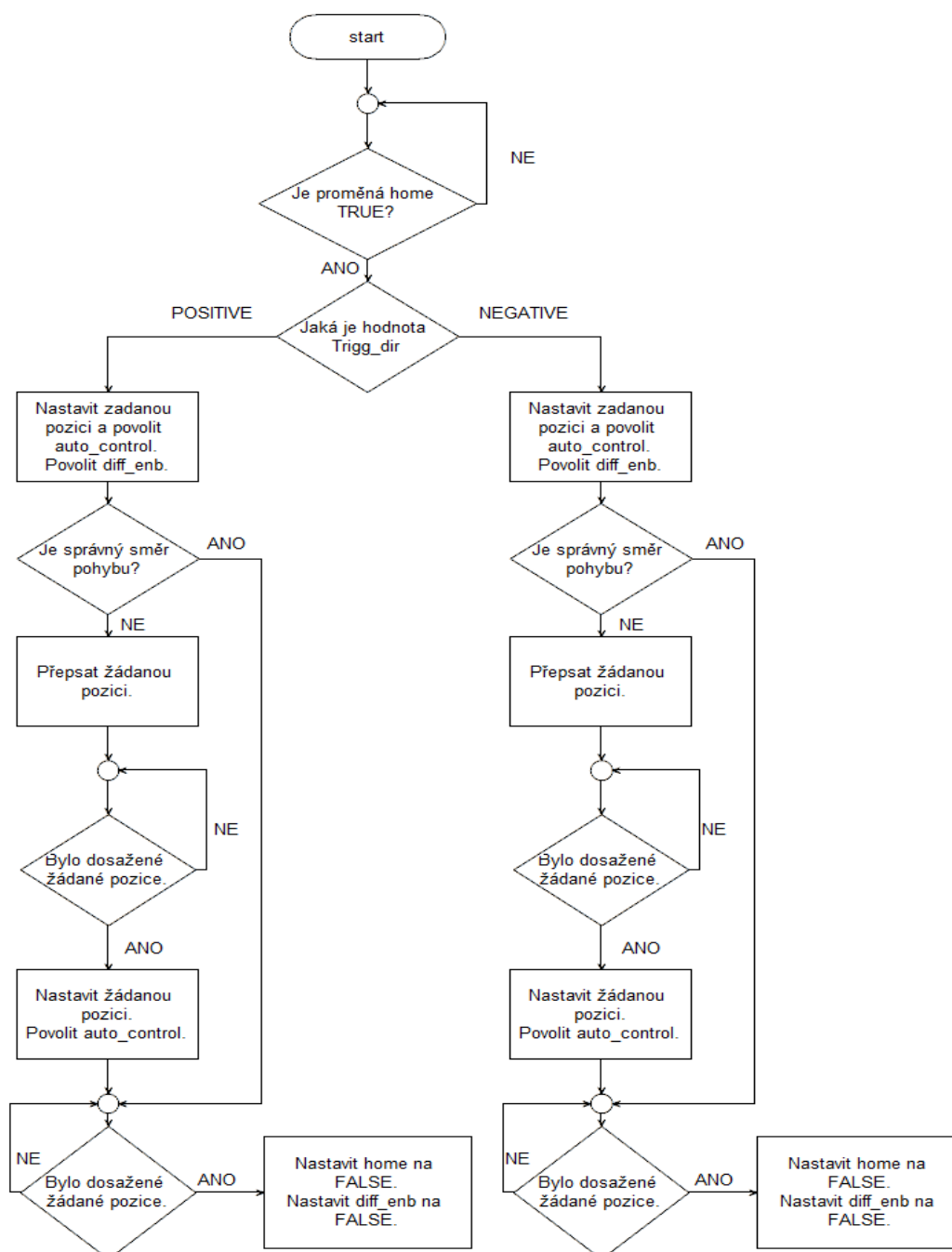


Obrázek 17 Vývojový diagram bloku pro výpočet difference

blok, jehož funkce však využívá většina ostatních bloků. Pokud je tento blok v chodu, dochází k ukládání hodnot v daných časových intervalech. Z těchto hodnot se poté počítá difference mezi časovými intervaly. Pokud tato difference splňuje určité požadavky, blok vyhodnotí, že bylo dosaženo žádané rychlosti, nebo že je možno provést reverzaci.

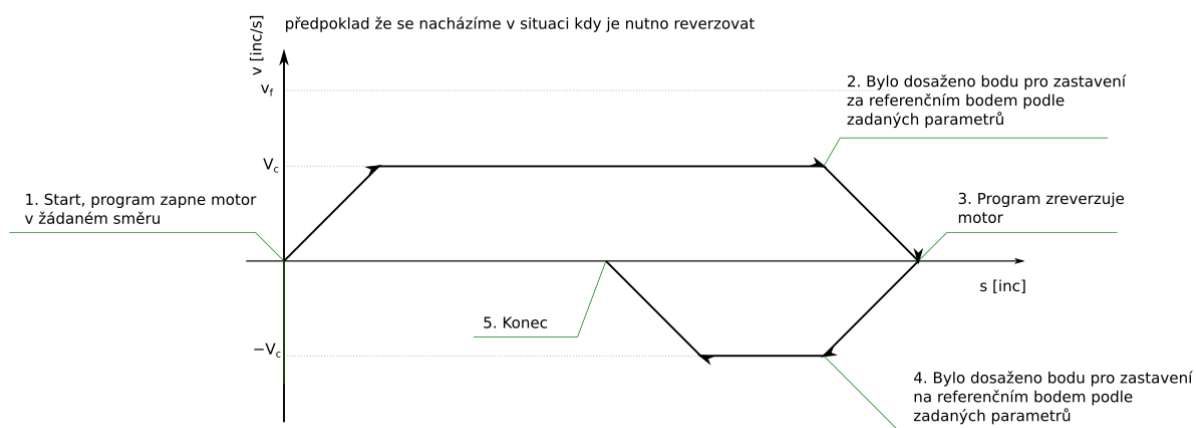
6.4 Blok „Home“

Úkolem tohoto bloku je provést nájezd na pozici, ta je nastavená buď jako pozice referenčního čidla nebo na



Obrázek 18 Vývojový diagram bloku pro nájezd na home pozici

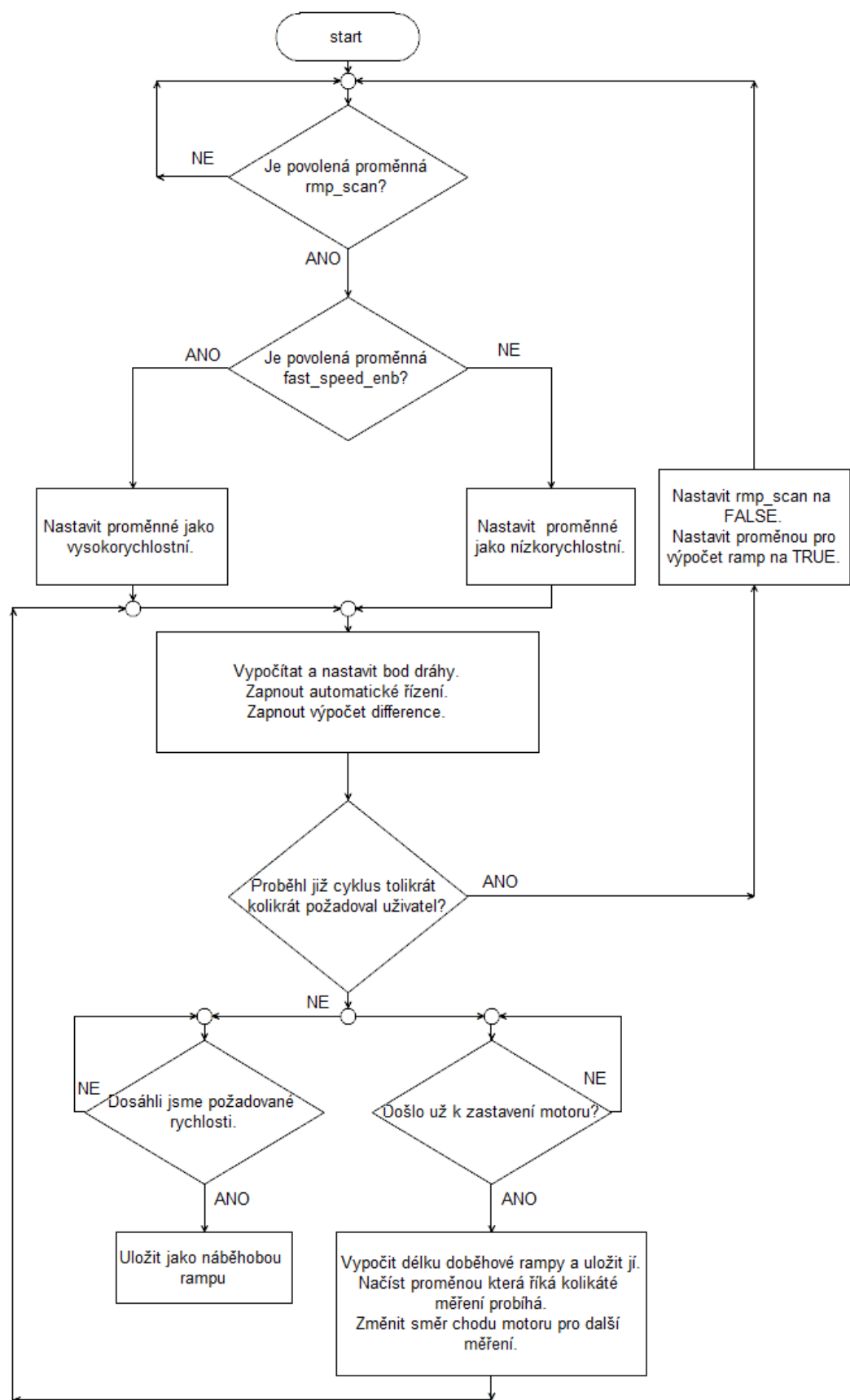
hodnotu zadanou uživatelem. Přes parametry je možno nastavit, zdali chceme na pozici najet pozitivním směrem nebo negativním. Pokud je požadavek na směr nájezdu odporuje směru, ve kterém se nachází pozice vůči relativní poloze. Dojde k tomu, že program bod přejede o uživatel definovanou vzdálenost zastaví provede reverzaci a najede zpět, rychlostní profil této situace popisuje Obrázek 19.



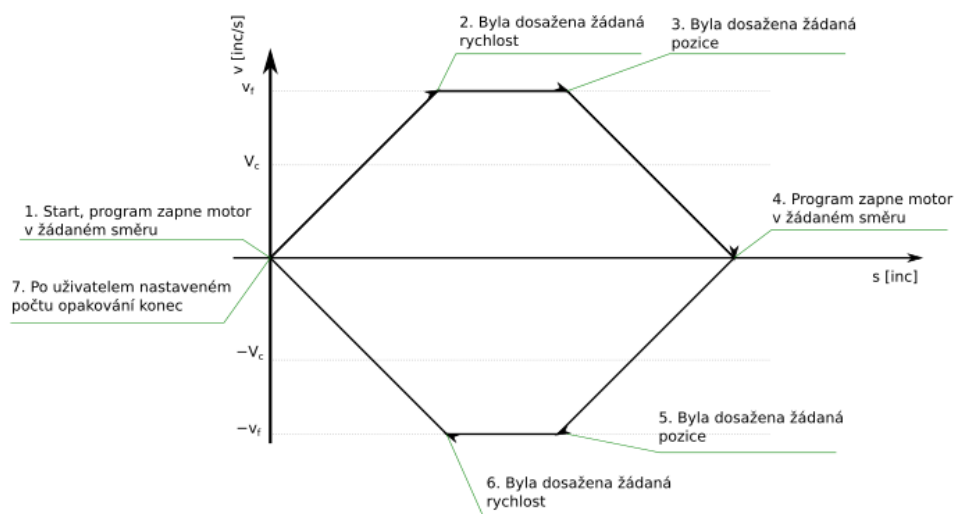
Obrázek 19 Rychlostní profil bloku pro nájezd na home pozici

6.5 Blok „RMP Scan“

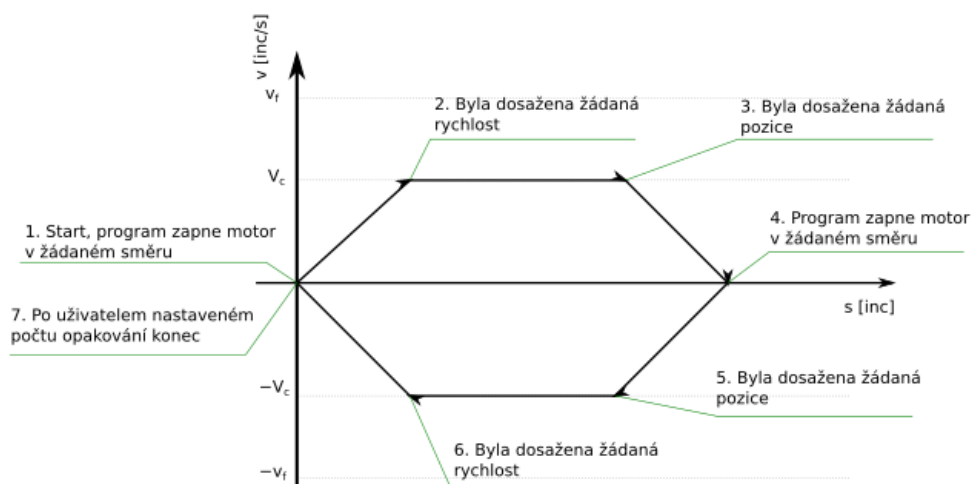
Tento blok provádí automatické měření ramp podle zadaných parametrů uživatele. Je nutno zadat počet požadovaných měření rampy, zdali se mají měřit rampy pro vysokou nebo nízkou rychlost, a nastaví také vzdálenost pro každé měření. Musí být tedy zaručeno že proměnná `M440.userdefined.ramp_jump` má dostatečnou velikost, aby motor stihl zrychlit na žádanou rychlost. Obrázek 22 zobrazuje průběh rychlostního profilu pro měření pomalých ramp. Obrázek 21 poté profil pro měření rychlých ramp.



Obrázek 20 Vývojový diagram bloku pro měření ramp



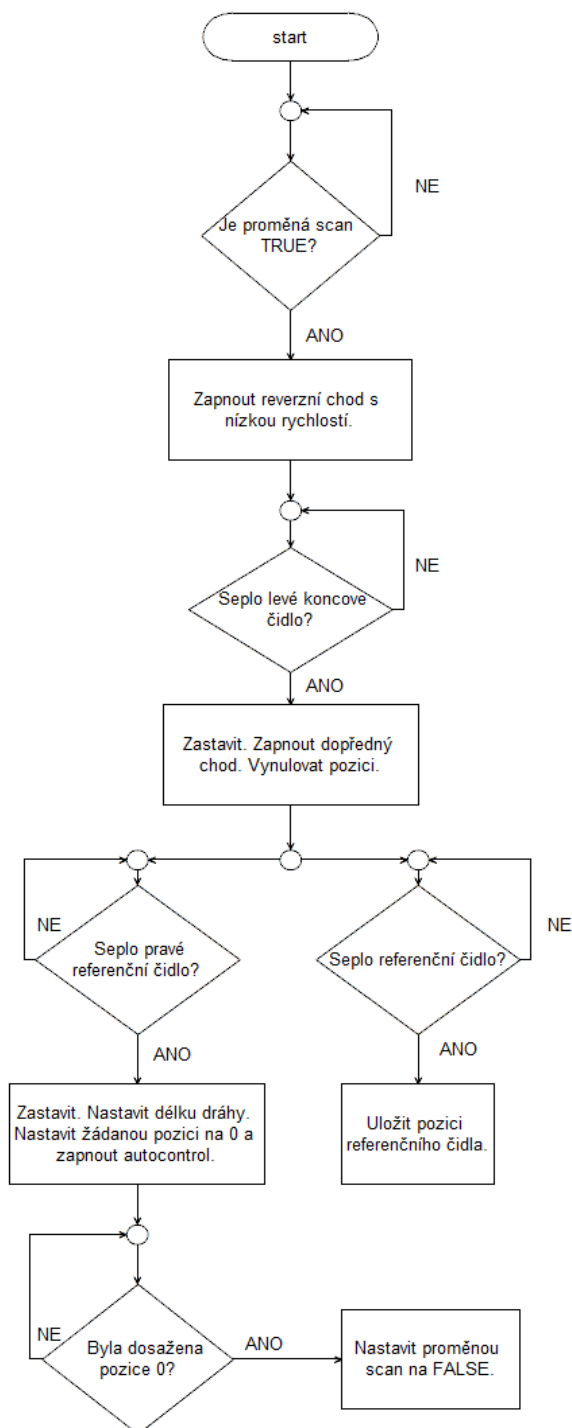
Obrázek 21 Rychlostní profil bloku pro měření ramp při vysoké rychlosti



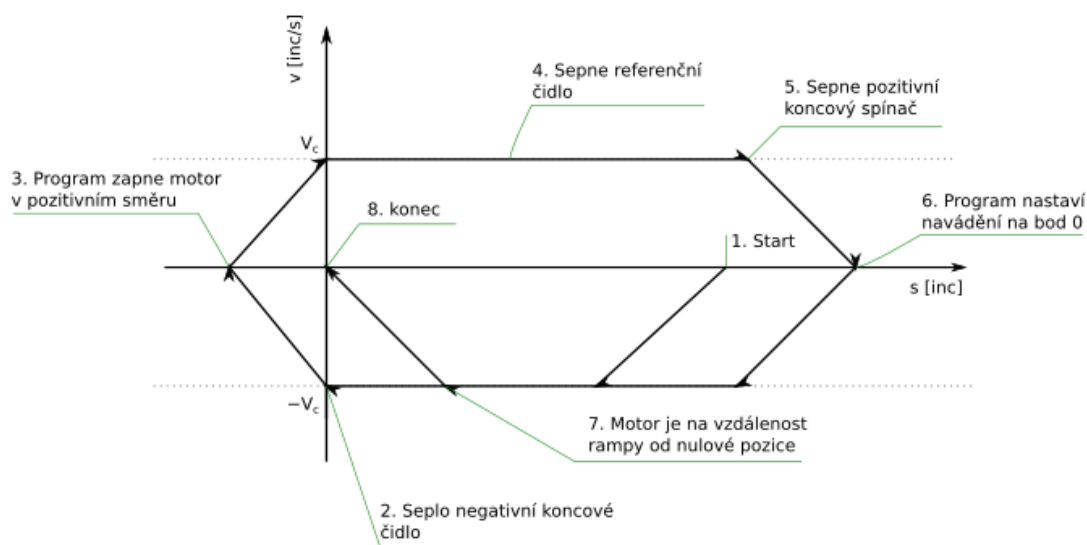
Obrázek 22 Rychlostní profil bloku měření ramp při nízké rychlosti

6.6 Blok „Scan“

Blok provádí automatický sken dráhy – provádí měření délky dráhy mezi záporným a kladným provozním koncovým snímačem. Tento blok nemá žádnou možnost uživatelského nastavení lze ho pouze zapnout, jakmile se tak stane provede se změření dráhy, rychlostní profil tohoto měření zobrazuje Obrázek 24.



Obrázek 23 vývojový diagram bloku sken



Obrázek 24 Rychlostní profil bloku pro sken dráhy

7 Náповěda

V této kapitole je popsáno uživatelské rozhraní mezi programem a nadřazeným uživatelským programem. Rozhraní je tvořeno strukturovanou proměnnou typu T_M440, která obsahuje níže uvedené dílčí podstruktury:

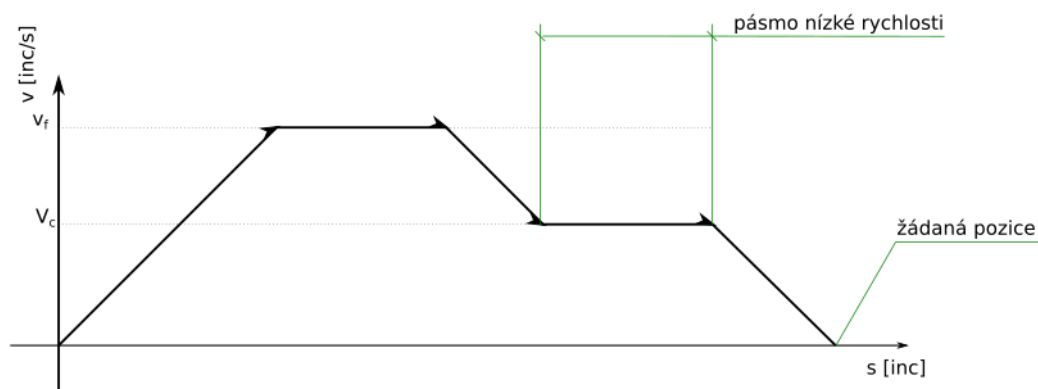
CMD	– Řídící příkazy
USER_DEFINED	– Uživatelem definovatelné proměnné
STATUS	– Informace o stavu a chodu motoru
HW	– Hardwarové vstupy a výstupy
REPORTS	– Krizová hlášení
INTERNAL	– Vnitřní proměnné

Obsah těchto dílčích podstruktur je uveden v následujících tabulkách.

7.1 Seznam důležitých proměnných pro uživatele:

Tabulka 2 Důležité proměnné

struktura	název	typ	význam	
CMD	start	BOOL	start v pozitivním směru	
	rev	BOOL	start v protichůdném směru	
	fast	BOOL	zadání vysoké rychlosti	
	slow	BOOL	zadání nízké rychlosti	
	Fault_ack	BOOL	potvrzení o detekci a odstranění chyby	
	stop	BOOL	povel k zastavení pokud je nastaven na true nelze ručně nastartovat motor	
	home	BOOL	příkaz k nájezdu na referenční čidlo	
	scan	BOOL	příkaz na zahájení funkce sken dráhy	
	auto_control	BOOL	povolení a zahájení automatického nájezdu na žádaný bod	
	rmp_scan	BOOL	příkaz na zahájení skenu ramp	
struktura	název	typ	význam	
USER DEFINED	konc_positive	DINT	10 = NO 20 = NC	
	konc_negative	DINT	10 = NO 20 = NC	
	ref	DINT	10 = NO 20 = NC	
	pasmo_necitlivosti	DINT	pásma necitlivosti pro senzor	
	pasmo_nizke_rychlosti	DINT	pro autocontrol	
	Trigg_dir	DINT	10 = pozitivní směr 20 = negativní směr	
	ramp_jump	DINT	vzdálenost pro jeden cyklus měření ramp	
	cycles	INT	počet cyklů pro měření ramp	
	rev_slow_ramp	DINT	Délka negativní pomalé rampy	Doběhové rampy
	rev_fast_ramp	DINT	Délka negativní rychlé rampy	
	fow_slow_ramp	DINT	Délka pozitivní pomalé rampy	
	fow_fast_ramp	DINT	Délka pozitivní rychlé rampy	
	rev_slow_rise_ramp	DINT	Délka negativní pomalé rampy	Náběhové rampy
	rev_fast_rise_ramp	DINT	Délka negativní rychlé rampy	
	fow_slow_rise_ramp	DINT	Délka pozitivní pomalé rampy	
	fow_fast_rise_ramp	DINT	Délka pozitivní rychlé rampy	
	poz_ref_cidla	DINT	pozice referenčního čidla	
	lenght	DINT	délka dráhy	
	zadana_pozice	DINT	žádaná pozice pro autocontrol	
	fast_speed_enb	BOOL	povolení vysoké rychlosti pro autocontrol	
	scan_fast_ramp	BOOL	sken ramp vysoké rychlosti	
	scan slow ramp	BOOL	sken ramp nízké rychlosti	



Obrázek 25 Zobrazení proměnné pásma nízké rychlosti

struktura	název	typ	význam
HW	DIN1	BOOL	FALSE = negativní koncový snímač aktivní
	DIN2	BOOL	FALSE = pozitivní koncový snímač aktivní
	DIN3	BOOL	FALSE = referenční snímač aktivní
	DOUT1	BOOL	TRUE = start v pozitivním směru
	DOUT2	BOOL	TRUE = start v negativním směru
	DOUT3	BOOL	Fast (*)
	DOUT4	BOOL	Slow (*)
			Digitální Vstupy
			Digitální výstupy

(*)Možné kombinace obou vstupů definuje Tabulka 6

struktura	název	typ	význam
STATUS	fault	BOOL	TRUE = nastala chyba
	move_positive	BOOL	TRUE = se motor pohybuje v kladném směru
	move_negative	BOOL	TRUE = se motor pohybuje v záporném směru
	home_set	BOOL	TRUE = program ví, kde se nachází referenční čidlo
	HW_error	BOOL	TRUE = nastala hardwarová chyba
	home	BOOL	TRUE = je aktivován blok HOME
	auto_control	BOOL	TRUE = je aktivován blok AUTO COTROL
	scan	BOOL	TRUE = je aktivován blok SCAN
	rmp_scan	BOOL	TRUE = je aktivován blok RMP SCAN

7.2 Nastavení měniče

7.2.1 Obecné nastavení měniče

1. Výběr frekvencí má být nastaven na výběr fixních frekvencí pomocí kombinace digitálních vstupů.
2. Zapínání motoru má být typem Zapnout, Zapnout Reverzace.
3. Hodnoty fixních frekvencí mají být:

Tabulka 3 Nastavení hodnot fixních frekvencí

0. fixní frekvence	0 Hz
1. fixní frekvence	nízká frekvence
2. fixní frekvence	vysoká frekvence

4. Nastavení digitálních výstupů má být:

Tabulka 4 Nastavení digitálních výstupů

DOUT1	ZAP / VYP1
DOUT2	ZAP (zpětný chod) / VYP1
DOUT3	výběr fixní frekvence viz. Tabulka 6
DOUT4	výběr fixní frekvence viz. Tabulka 6

5. Rampy měniče mají být nastaveny na žádanou hodnotu.

7.2.2 Nastavení měniče M440

Tabulka 5 Nastavení parametrů měniče M440

parametr	hodnota	význam
P1000	3	nastavení frekvence na pevné kmitočty
P0701	1	nastavení DOUT1 na ZAP / VYP1
P0702	2	nastavení DOUT 2 na ZAP (zpětný chod) / VYP1
P0703	15	nastavení DOUT 3 na výběr fixní frekvence
P0704	15	nastavení DOUT 4 na výběr fixní frekvence
P1001	0	nulová fixní frekvence je nula
P1002	žádaná nízká frekvence	
P1003	žádaná vysoká frekvence	
P1020	nastavení náběhové rampy	
P1021	nastavení doběhové rampy	

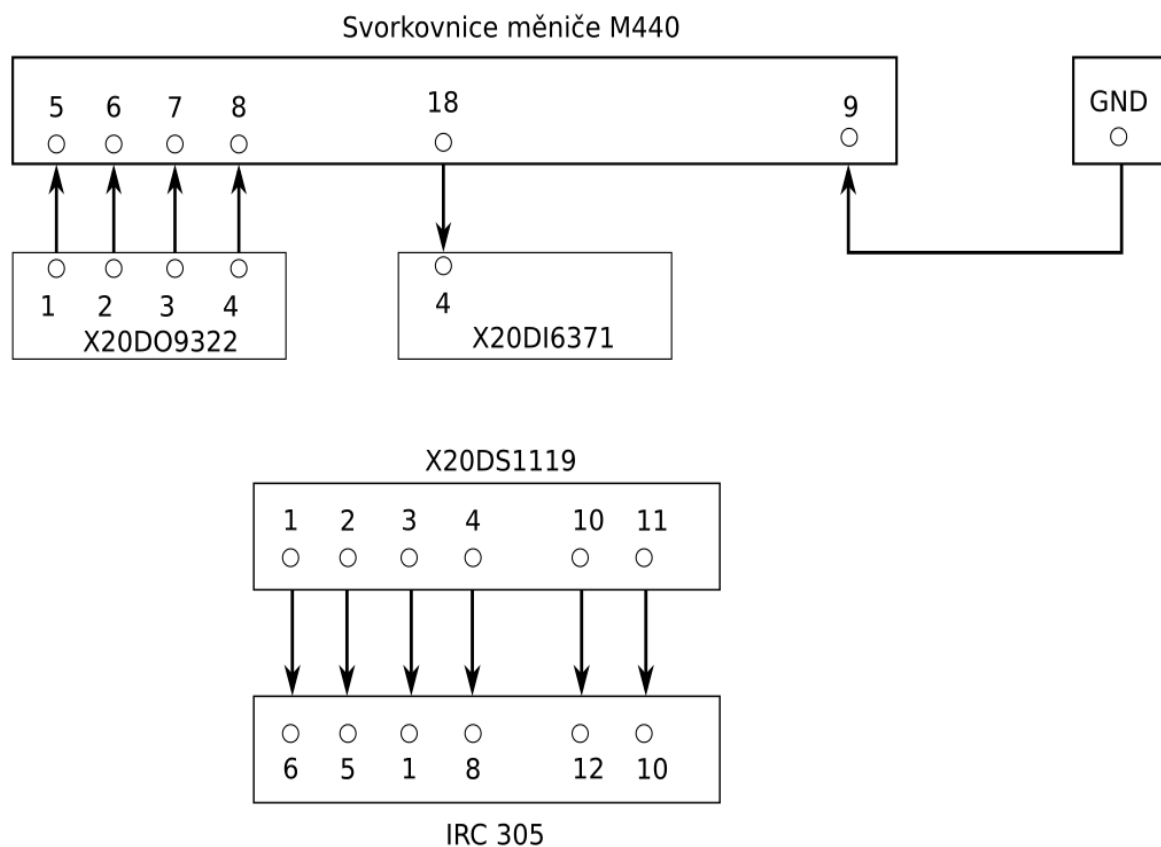
Tabulka 6 Kombinace výstupů pro zadání frekvencí

DOUT 3	DOUT 4	výstup
0	0	0 Hz
0	1	Nízká frekvence
1	0	Vysoká frekvence

Zbylé parametry je doporučeno nastavit podle manuálu k měniči a před nastavením je záhodno provést reset měniče do továrního nastavení.

7.3 Zapojení úlohy

Následující schéma (Obrázek 26) zachycuje signálové propojení jednotlivých komponent testovacího pracoviště. Koncové snímače byly v naší úloze pouze virtuální, pokud by však měly být připojeny byly by napojeny do prvních tří slotů modulu X20DI6371.



Obrázek 26 Signálové zapojení úlohy

Pro modul X20DS1119 byly zapojeny dva kanály IRC čidla a jejich negace. Napájení čidla je zajištěno modulem X20DS1119 +5V je port 10 a GND je port 11.

Tabulka 7 Seznam propojení svorkovnic

	Číslo svorky	Název svorky	Připojení	Význam signálu
X20DO9322	1	DO1	M440: 5	povel k chodu v kladném směru
	2	DO2	M440: 6	povel k chodu v záporném směru
	3	DO3	M440: 7	výběr 1. (pomalé) fixní frekvence
	4	DO4	M440: 8	výběr 2. (rychlé) fixní frekvence
	5	DO5	x	
	6	DO6	x	
	7	DO7	x	
	8	DO8	x	
	9	DO9	x	
	10	DO10	x	
	11	DO11	x	
	12	DO12	x	
	Číslo svorky	Název svorky	Připojení	Význam signálu
X20DI6371	1	DI1	x	
	2	DI2	x	
	3	DI3	x	
	4	DI4	M440 : 18	Chyba měniče
	5	DI5	x	
	6	DI6	x	
	7	+24VDC	x	
	8	+24VDC	x	
	9	+24VDC	x	
	10	+24VDC	x	
	11	+24VDC	x	
	12	+24VDC	x	
	Číslo svorky	Název svorky	Připojení	Význam signálu
X20DS1119	1	Ch1	IRC: 5	kanál 1 z čidla IRC
	2	Ch1\	IRC: 6	negovaný kanál 1 z čidla IRC
	3	Ch2	IRC: 8	kanál 2 z čidla IRC
	4	Ch2\	IRC: 1	negovaný kanál 2 z čidla IRC
	5	Ch3	x	
	6	Ch3\	x	
	7	Ch4	x	
	8	Ch5	x	
	9	+24VDC	x	
	10	+5VDC	IRC: 12	napájení čidla IRC 305 +5V
	11	GND	IRC: 10	napájení čidla IRC 305 GND
	12	GND	x	

	Číslo svorky	Název svorky	Připojení	Význam signálu
M440	1	+10 V	x	
	2	0V	x	
	3	ADC1+	x	
	4	ADC1-	x	
	5	DIN1	DO9322 :1	povel k chodu v kladném směru
	6	DIN2	DO9322 :2	povel k chodu v záporném směru
	7	DIN3	DO9322 :3	výběr 1. (pomalé) fixní frekvence
	8	DIN4	DO9322 :4	výběr 2. (rychlé) fixní frekvence
	9	Output +24v	x	
	10	ADC2+	x	
	11	ADC2-	x	
	12	DAC1+	x	
	13	DAC1-	x	
	14	PTCA	x	
	15	PTCB	x	
	16	DIN5	x	
	17	DIN6	x	
	18	NC RELAY1	x	
	19	NO RELAY1	x	
	20	COM RELAY1	x	
	21	NO RELAY2	x	
	22	COM RELAY2	x	
	23	NC RELAY3	x	
	24	NO RELAY3	x	
	25	COM RELAY3	x	
	26	DAC2+	x	
	27	DAC2-	x	
	28	Output 0v	GND	
	29	P+	x	
	30	N-	x	

7.4 Návod ke spuštění jednotlivých bloků.

Zde jsou shrnuty všechny nápovědy k jednotlivým blokům, stejnou nápovědu lze nalézt v programu samém tam je však každá z nich přiřazena k příslušnému bloku.

Tabulka 2 obsahuje seznam důležitých proměnných.

7.4.1 Prvotní spuštění - inicializace

- I. Nastavit typy čidel a parametr pro nájezd na referenční čidlo. Tabulka 2 obsahuje seznam důležitých proměnných. (jedná se o proměnné `konc_positive`, `konc_negative` a `ref`)
- II. Spustit sken ramp nebo nastavit délky ramp ručně v inicializační části programu.
- III. Spustit sken dráhy nebo nastavit parametr `M440.internal.lenght` a `M440.internal.poz_ref_cidla` ručně v inicializační části programu.

7.4.2 Blok pro automatické nájezdy na bod

- I. `M440.userdefined.zadana_pozice` je požadovaný bod.
- II. `M440.userdefined.fast_speed_enb` povolí bloku užít vysokou rychlost.
- III. `M440.userdefined.pasmo_nizke_rychlost` nastaví jakou část dráhy před zastavením má motor jet na nízké rychlosti.
- IV. `M440.CMD.auto_control` spustí proces nájezdu na bod.

7.4.3 Blok pro sken dráhy

- I. `M440.CMD.scan` spustí sken, pokud nechceme provést funkci je nutno nastavit proměnné `M440.userdefined.poz_ref_cidla` jako pozici referenčního čidla a `M440.userdefined.lenght` jako délku dráhy.

7.4.4 Blok pro nájezd na referenční čidlo

- I. Je nutno nastavit proměnnou `M440.userdefined.Trigg_dir` ta určuje v jakém směru chceme dojet na referenční čidlo.
- II. Je nutno nastavit proměnnou `M440.userdefined.presah` ta určuje o kolik se má přejet čidlo, než dojde k reverzaci. Pokud ho chceme spustit z druhé strany než se nachází aktuální pozice motoru.
- III. `M440.CMD.home` spustí proces nájezdu na referenční čidlo

7.4.5 Blok pro sken délky ramp

- I. `M440.userdefined.fast_fast_ramp` nastaví sken na rampy pro vysokou rychlost. `M440.userdefined.scan_slow_ramp` nastaví sken na rampy pro nízkou rychlost.
- II. Nastavit `M440.userdefined.cycles` podle toho kolikrát chceme provést měření.
- III. Nastavit `M440.userdefined.ramp_jump` podle toho o kolik se má motor posunout při každém měření.
- IV. `M440.CMD.rmp_scan` spustí proces skenu ramp.
- V. Naměřené rampy je vhodné zanést do inicializační části `Program Init.st`.

Pokud chce uživatel zadat proměnné ručně je záhodno tak učinit v inicializační části programu. Tím docílíme toho, že se hodnoty uloží a zůstanou uloženy i po případném restartu.

8 Ověření funkčnosti

Navržený program byl testován pro různá nastavení uživatelských parametrů, bylo ověřeno chování jednotlivých funkčních bloků. Následně jsme provedli opakovaná měření rozběhových a zejména doběhových ramp a měření přesnosti polohování.

8.1 Postup a způsob měření:

Pro získání hodnot ramp byl použit následující postup:

Na měniči byla nastavena příslušná rampa, v programu byly hodnoty ramp nastaveny na nulovou hodnotu, v programu byla nastavena hodnota, na kterou má motor najet. Díky tomu že program má nastavené nulové rampy dojde k vydání povelu zastavení až v bodě, kdy motor najede na zadaný bod. O kolik motor tento bod motor přejede, je poté vyhodnoceno jako délka rampy. Pro zaznamenání nájezdu na bod byla využita vnitřní funkce *Trace* programu AS, která umožňuje sledovat a ukládat hodnoty žádných proměnných v závislosti na čase.

Měření bylo několikrát opakováno. Výsledná hodnota, se kterou má program pracovat je průměrná hodnota z takto naměřených hodnot.

Na konci měření bylo provedeno druhé měření ramp, a to u nízké rychlosti. V grafech jsou to výsledky označené zelenou barvou. Důležité je také zmínit, že v grafech je potlačena nula.

8.2 Tabulka naměřených hodnot ramp

Tabulka obsahuje opakované měření délky dojezdových ramp při různých směrech pohybu (negative, positive) a různých rychlostech pohybu (slow, fast). Pro slow rampy bylo měření provedeno dvakrát jednou na hned po spuštění motoru, podruhé když byl motor už nějakou dobu v chodu. Levá tabulka je pro sadu měření při pomalé rychlosti pravá pak pro sadu měření při vysoké rychlosti.

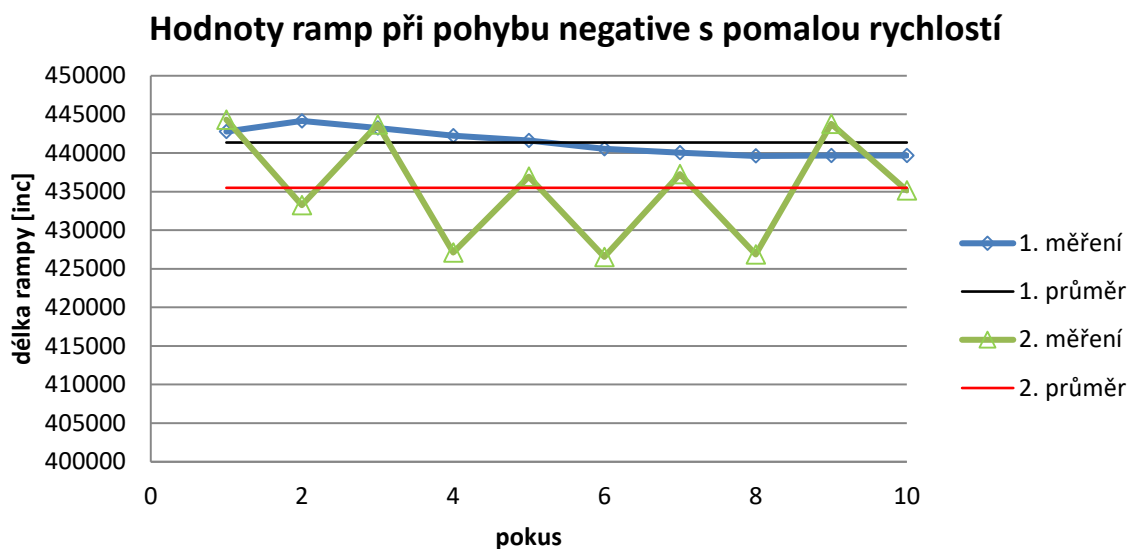
Tabulka 8 Naměřené hodnoty ramp

pokus	Délka rampy [inc]			
	Pozitivní směr Slow		Negativní směr Slow	
	1. měření	2. měření	1. měření	2. měření
1	443092	427324	442783	444318
2	444433	438462	444152	433258
3	444142	426806	443233	443671
4	445353	440681	442223	427119
5	426341	429613	441624	436922
6	427440	425886	440533	426538
7	427867	434043	440029	437248
8	428106	444058	439609	426849
9	428269	432245	439683	443773
10	428336	441515	439650	435145

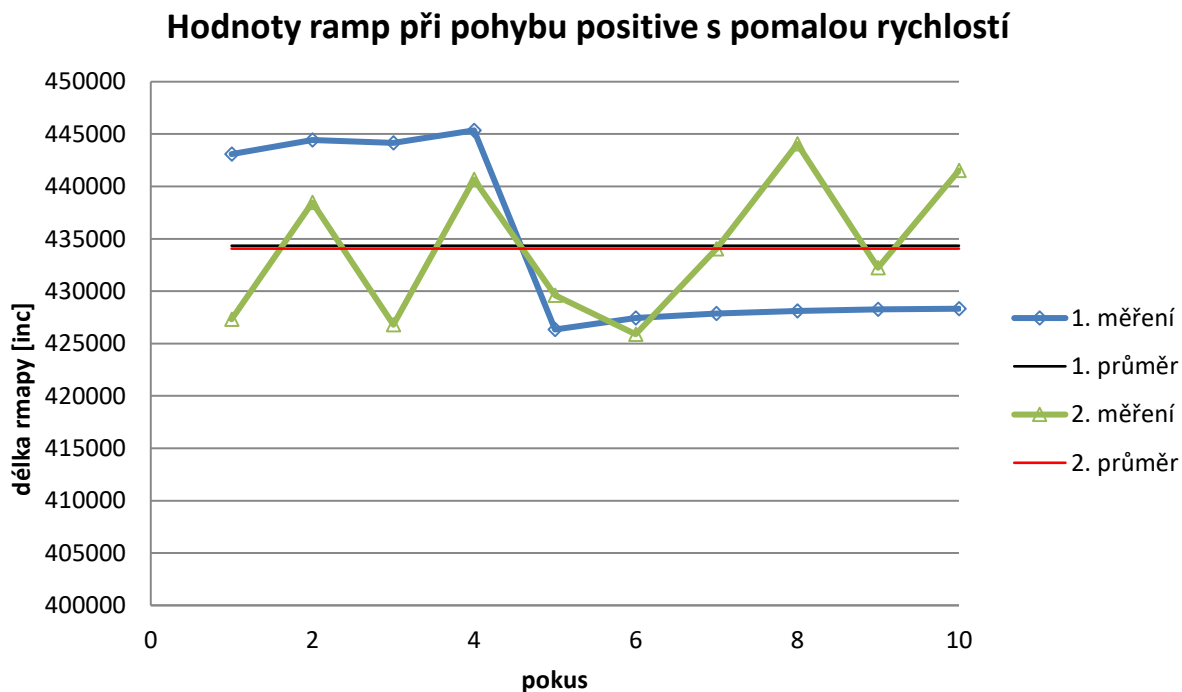
pokus	Délka rampy [inc]	
	Pozitivní směr fast	Negativní směr fast
1	959569	960359
2	962510	958118
3	965754	955169
4	958807	955248
5	960415	953564
6	962146	953266
7	960128	952908
8	959560	953989
9	960119	953975
10	959596	954222

8.3 Grafy naměřených hodnot

Následující grafy zobrazují hodnoty z předcházejících tabulek. Do grafů je vynesena i hodnota průměrných délek naměřených ramp. V této podkapitole se jedná o grafy pro nízkou rychlost u nich bylo měření provedeno dvakrát poprvé hned po startu motoru, po druhé, když už motor pracoval po delší dobu.



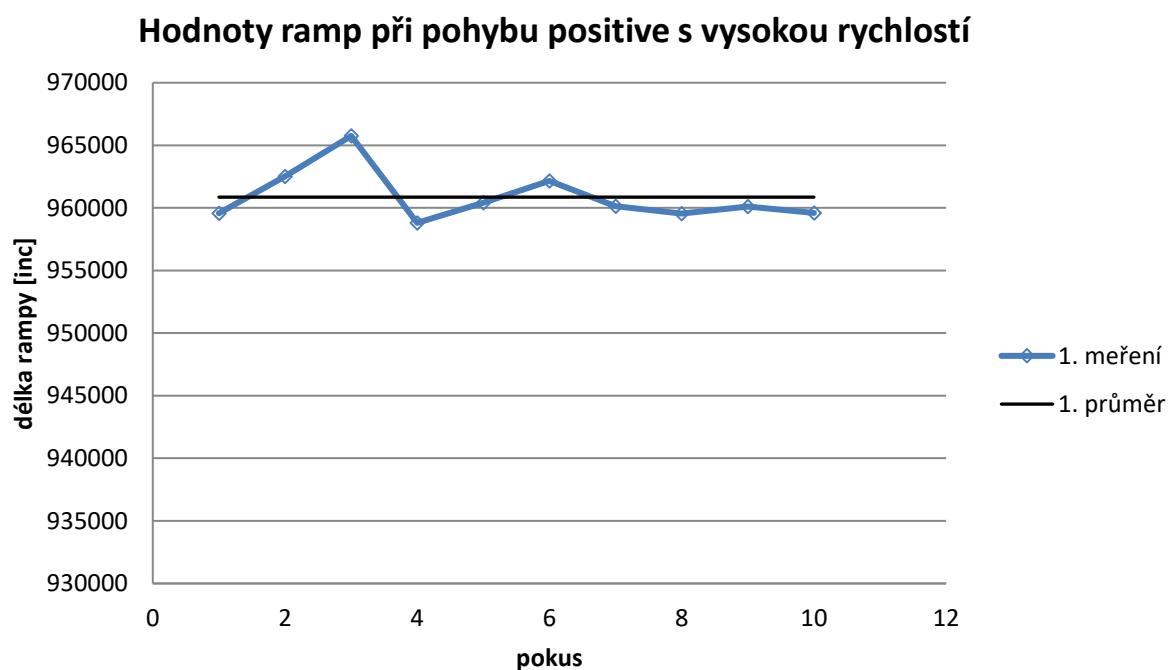
Graf 1 měření ramp při nízké rychlosti a negativním směru



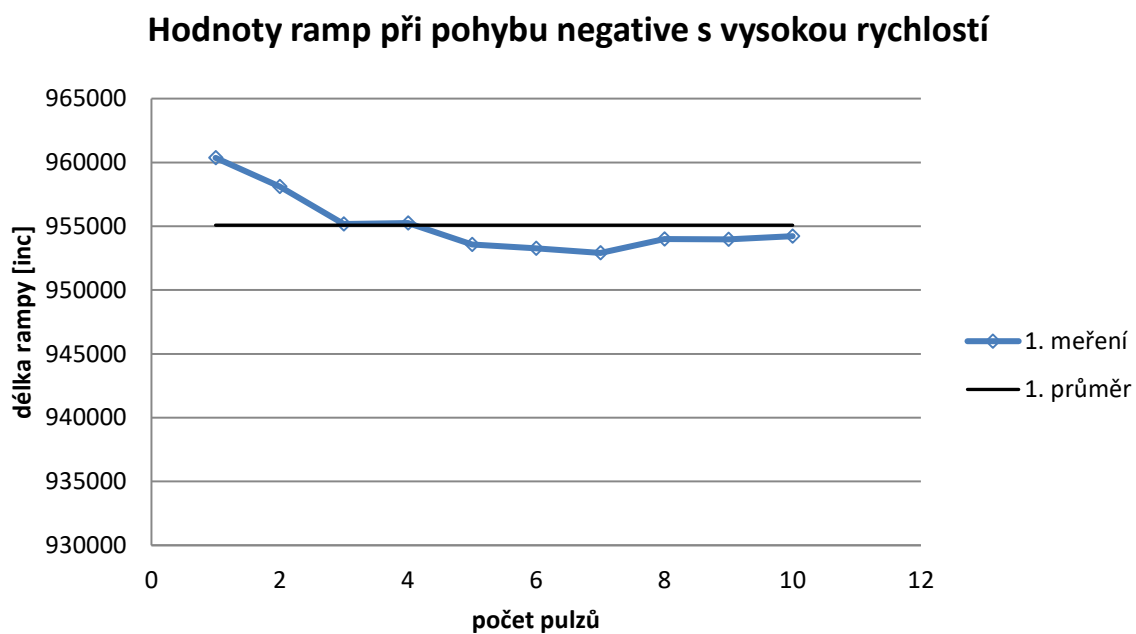
Graf 2 měření ramp při nízké rychlosti a pozitivním směru

8.4 Měření ramp vysoké rychlosti

V této podkapitole se jedná o grafy pro vysoké rychlosti.



Graf 3 měření ramp při vysoké rychlosti a pozitivním směru



Graf 4 měření ramp při vysoké rychlosti a negativním směru

8.5 Polohování s nastavenými rampami

Tabulka 9 Polohování v negativním směru

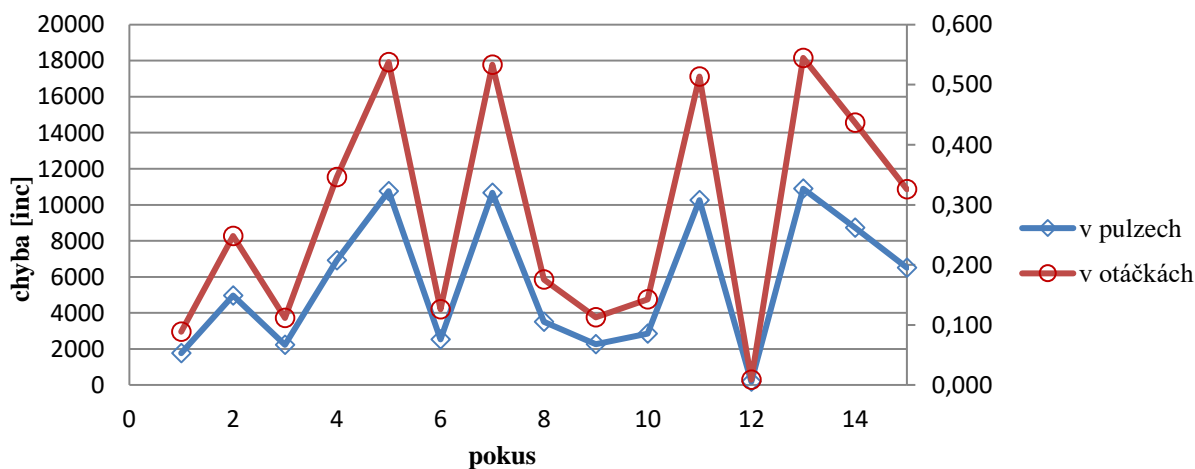
Negativní směr				
pokus	žádaná pozice [inc]	Skutečná pozice [inc]	Chyba [ot]	Absolutní chyba [inc]
1	0	-1769	0,09	1769
2	0	-4964	0,25	4964
3	-2000000	-1997774	0,11	2226
4	-8000000	-7993081	0,35	6919
5	2000000	2010753	0,54	10753
6	0	-2523	0,13	2523
7	0	10668	0,53	10668
8	1000000	1003513	0,18	3513
9	-2000000	-2002257	0,11	2257
10	-1800000	-1802845	0,14	2845
11	-4000000	-3989733	0,51	10267
12	-1000000	-1000177	0,01	177
13	8000000	8010893	0,54	10893
14	5500000	5508735	0,44	8735
15	1000000	1006517	0,33	6517

Výpočet chyby měření v otáčkách.

$$Chyba = \frac{||\text{žádaná pozice}| - |\text{skutečná pozice}||}{20\,000} \quad (3)$$

Na následujících grafech můžeme vidět, jaké přesnosti polohování bylo dosažení při nastavení ramp podle výsledků předchozích měření.

Chyba polohování při negativním pohybu

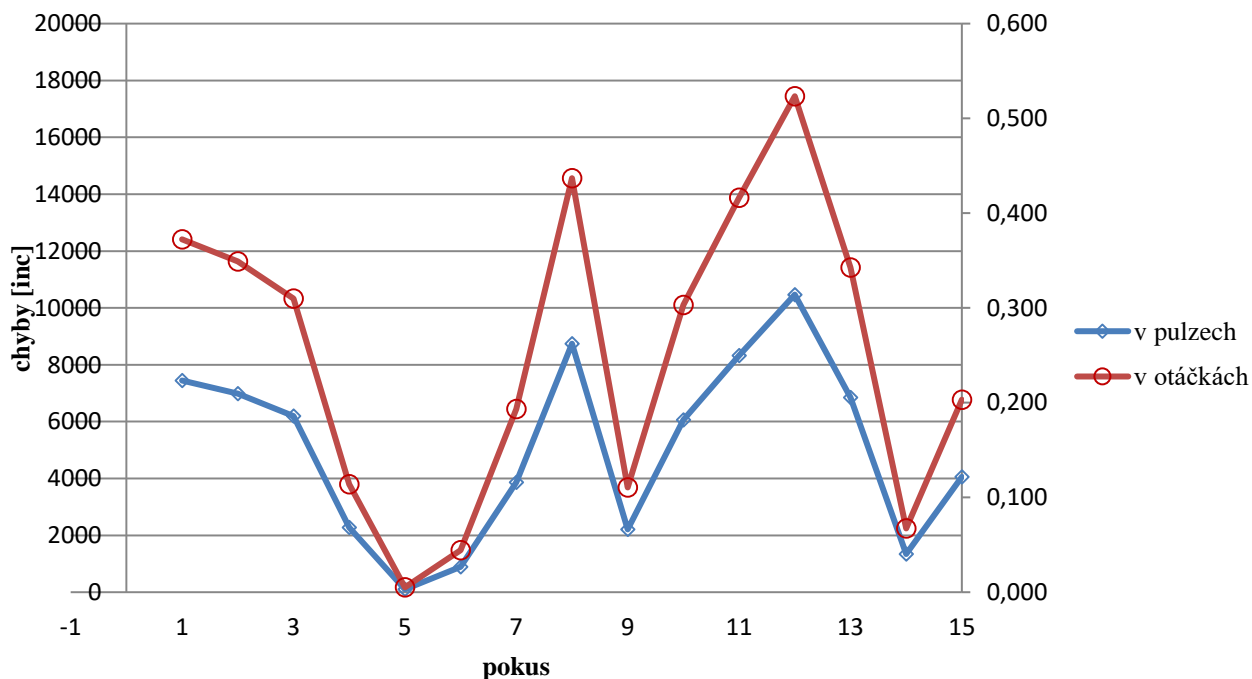


Graf 5 Měření přesnosti polohování při negativním směru

Tabulka 10 Polohování v pozitivním směru

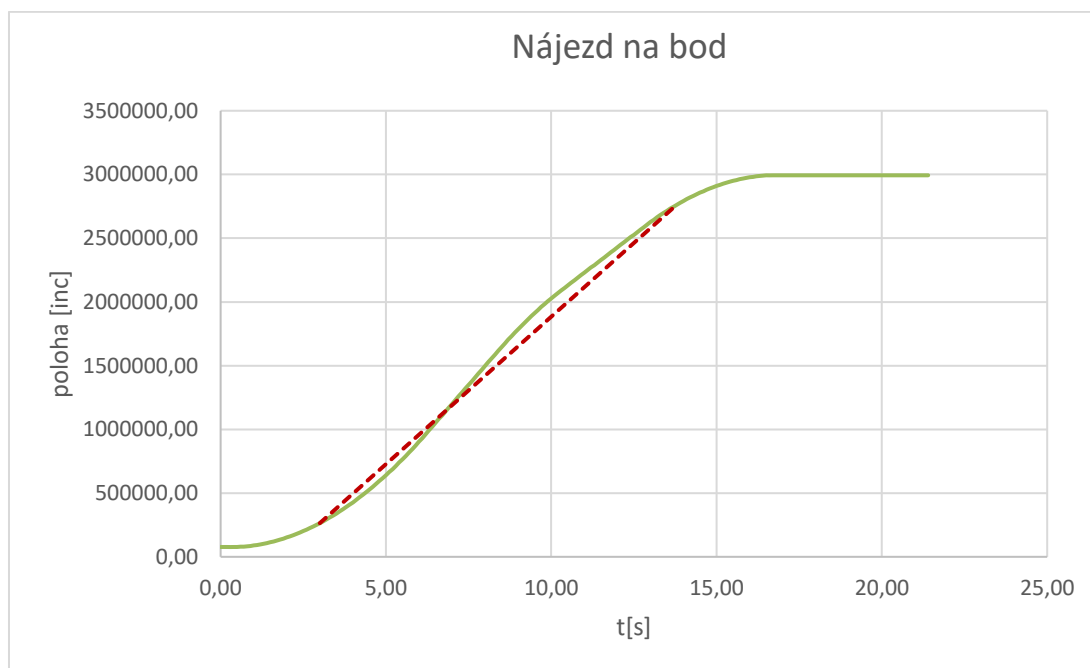
Pozitivní směr				
pokus	žádaná pozice [inc]	výsledek [inc]	Chyba [ot]	chyba [inc]
1	3000000	3007447	0,37	7447
2	3000000	2993017	0,35	6983
3	4500000	4493801	0,31	6199
4	2500000	2502278	0,11	2278
5	-2000000	-1999898	0,01	102
6	5000000	4999112	0,04	888
7	2500000	2496136	0,19	3864
8	3000000	2991264	0,44	8736
9	5000000	4997792	0,11	2208
10	0	6065	0,30	6065
11	-1000000	-1008326	0,42	8326
12	-5000000	-4989533	0,52	10467
13	-3000000	-3006854	0,34	6854
14	0	1345	0,07	1345
15	10000000	9995941	0,20	4059

Chyba polohování při pozitivním pohybu



Graf 6 Měření přesnosti polohování při pozitivním směru

8.6 Graf nájezdu na žádanou pozici



Graf 7 Vývoj pozice při nájezdu na žádaný bod

8.7 Výsledky měření

S rampami, které jsme naměřili, bylo dosaženo přesnosti polohování 0,6 otáčky, což je vyhovující. Zajímavých výsledků bylo dosaženo při opětovném měření ramp, které bylo provedeno na konci měření. Jak je vidět z grafu, rozptyl těchto hodnot je velmi odlišný od hodnot, které byly získány na začátku měření. Dle mého názoru je to způsobeno dobou chodu motoru a tím, že se během našeho měření zahřál a došlo k lehké změně parametrů vůči začátku měření, kde byl motor zcela nevyužíván. U ramp v dopředném chodu se tato změna nijak výrazně neprojevila. U zpětného chodu motoru byla změna již výraznější – došlo k rozdílu průměrů naměřených hodnot o 5868 pulzů (něco přes $\frac{1}{4}$ otáčky) což, jak je vidět z grafu přesnosti polohování, způsobilo i větší chybu přesnosti polohování při zpětném chodu. Ani tak však nedošlo k větší nepřesnosti, než je 0,6 otáčky. Při korekci rampy podle nové hodnoty došlo k lehkému zlepšení přesnosti. Co se týče nájezdu na bod je vidět že ani pokud pomineme zrychlení a zabrzdění a bereme pouze část nájezdu při konstantní rychlosti, ani tak není vývoj polohy v čase lineární, při zrychlení i zpomalení je vývoj polohy v čase silně nelineární. Měření náběhových ramp bylo sice provedené také ale v naší aplikaci byly rampy náběhové a doběhové téměř totožné (rozdíl byl v rámci půl až jednoho procenta) proto nejsou jejich hodnoty zavedeny v tabulkách a grafem pouze by došlo k zhoršení přehlednosti.

9 Možnosti budoucí modifikace programu

9.1 Modifikace pro rotační osu

Tuto modifikaci by nejlépe bylo možné realizovat přidáním další proměnné modulu. U ní by bylo vyhodnoceno, zdali je nenulová. Pokud ano, program předpokládá rotační osu, jinak se počítá s lineární osou. Co se významu

proměnné modulo týče, ta nám označuje počet inkrementů, které se načtou, než dojde k opětovnému dosažení výchozí pozice. Další otázkou by bylo, jakým způsobem by mělo být provedeno polohování na žádaný bod – zdali stylem, kdy se zvolí nejkratší možná dráha k dosažení bodu, nebo se vezme uživatelem požadovaný směr otáčení.

9.2 Modifikace pro zápornou polohu čidla

Podmínky, které program má pro určování směrů, výpočty brzdných bodů a podobně jsou dělané tak, aby byla jejich funkčnost zaručena i v případě, že se pohybujeme v záporných hodnotách. Program však ve výchozí podobě neumožňuje dosažení záporných hodnot, negativní koncový snímač se ukládá jako pozice 0 a pozitivní se ukládá jako maximální. Rozšíření programu o tuto možnost by šlo provést, ale vyžadovalo by lehký zásah do kódu. Pro měření výsledků bylo navádění do záporných poloh i otestováno, jak vyplývá z výše uvedených tabulek, na toto testování byly však vypnuty bezpečnostní prvky.

9.3 Zakázání polohování pro malé změny polohy

Pokud je zadán příkaz na polohování a žádaná poloha je nižší, než je poloha, u které je zajištěno že dojde k plnému zrychlení motoru dojde k polohování s velkou chybou. Je to způsobeno tím, že program počítá s lineárním tvarem rampy, tento předpoklad je však mylný, jak vyplývá i z měření, nicméně i když dojde k opravě tvaru rampy, vlivem skutečných vlastností motoru bude polohování nepřesné. Jako řešení je nejvhodnější takové, které používají i jiné realizace, a to je právě zákaz polohování s takto krátkou dráhou.

10 Závěr:

Po seznámení se s problémem polohování byla sestavena úloha, na které poté bylo možné vytvořit a otestovat program. Největším problémem bylo stanovení délek ramp, nakonec se povedlo naprogramovat blok, který je schopen délky těchto ramp naměřit. Dalším měřením se poté ověřila jeho funkčnost, když byly takto naměřené hodnoty rampy použity při polohování. Během něj bylo dosaženo dostatečné přesnosti (pod 0,6 otáčky).

K bloku byla následovně vytvořena nápověda, ve které se pojednává nejen o obecném nastavení, které je nutno provést, ale část se věnuje i variantě kde je použit stejný hardware. Dále je poskytnuta stručná nápověda ke každému důležitému programovému bloku stejně tak jako seznam důležitých proměnných s komentářem.

Jako možnosti dalšího rozvoje by se nabízela úprava, která by umožnila pracovat i s rotační osou, ne pouze lineární. Dále by se program dal rozšířit o možnost libovolně si posouvat začátek dráhy, který je nyní pevně definován jako nulový a zákaz polohování, když je požadavek příliš malý a dochází tak k větší nepřesnosti. Podrobněji jsou možnosti rozšíření rozebrány v kapitole 9.

V příloze lze nalézt finální verzi programu, elektronickou verzi této práce a nápovědu.

11 Seznam použité literatury

- [1] SIEMENS s.r.o. *Siemens Micromaster 440: Návod k obsluze a údržbě, verze C2*. 2003.
Dostupné z :
http://stest1.etnetera.cz/ad/current/content/data_files/technika_pohonu/menice/stridave_menice/nizkonepetove_menice/micromaster/micromaster_440/manualy/opi_micromaster_440_03-2003_cz.pdf
- [2] Datasheet X20(c)DI6371. *Https://www.br-automation.com/cs/* [online]. Brno: B&R, ©2018 [cit. 2018-12-17]. Dostupné z: https://www.br-automation.com/downloads_br_productcatalogue/BRP44400000000000000567926/X20DI6371-ENG_V3.17.pdf
- [3] Datasheet X20(c)DO9322. *Https://www.br-automation.com/cs/* [online]. Brno: B&R, ©2018 [cit. 2018-12-17]. Dostupné z: https://www.br-automation.com/downloads_br_productcatalogue/BRP44400000000000000564982/X20DO9322-ENG_V3.15.pdf
- [4] Datasheet X20(c)DS1119. In: *Https://www.br-automation.com/cs/* [online]. Brno: B&R, ©2018 [cit. 2018-12-17]. Dostupné z: https://www.br-automation.com/downloads_br_productcatalogue/BRP44400000000000000535810/X20DS1119-ENG_V3.35.pdf
- [5] Incremental rotary encoders IRC300 – 325. *Http://www.larm.cz/* [online]. Netolice: LAMR [cit. 2018-12-17]. Dostupné z: http://www.larm.cz/inkrementalni-rotacni-snimace-irc-300-az-325_files/
- [6] Návod k programu automation studio. *Https://www.br-automation.com/cs/* [online]. Brno: B&R, ©2018 [cit. 2018-12-18]. Dostupné z: <https://www.br-automation.com/cs/soubory-ke-stazeni/software/automation-studio/automation-studio-42/automation-studio-v42/>
- [7] Linear Positioning Module. *Https://www.rockwellautomation.com/cs_CZ/overview.page* [online]. Rockwell Automation, ©2018 [cit. 2018-12-18]. Dostupné z: https://literature.rockwellautomation.com/idc/groups/literature/documents/um/1771-um654_-en-p.pdf
- [8] FM 354 Positioning Module for Servo Drives. *Https://new.siemens.com/global/en.html* [online]. Siemens, ©SiemensAG2009-2018 [cit. 2018-12-18]. Dostupné z: <https://support.industry.siemens.com/cs/document/60676781/simatic-fm-354-positioning-module-for-servo-drives?dti=0&lc=en-CZ>
- [9] OPERATION MANUAL SYSMACPosition Control Units. *Https://www.omron.com/* [online]. OMRON, ©CopyrightOMRONCorporation2007-2018 [cit. 2018-12-18]. Dostupné z: http://www.omron.com.au/data_pdf/mnu/w426-e1-11_cj1w-nc_cs1w-nc.pdf?id=1758
- [10] Type QD70 Positioning Module User's Manual. *Https://www.mitsubishielectric.com/en/index.html* [online]. Mitsubishi Electric Corporation, ©MitsubishiElectricCorporation [cit. 2018-12-18]. Dostupné z: <http://dl.mitsubishielectric.com/dl/fa/document/manual/plc/sh080171/sh080171n.pdf>

12 Přílohy